NAVAL POSTGRADUATE SCHOOL

Monterey, California

DISSERTATION

**A SOFTWARE TECHNOLOGY TRANSITION
ENTROPY BASED ENGINEERING MODEL**

by

Michael S. Saboe

March 2002

Dissertation Supervisor: Luqi

Approved for public release; distribution is unlimited

THIS PAGE INTENTIONALLY LEFT BLANK

REPORT DOCUMENTATION PAGE

Form Approved OMB No. 0704-0188

Public reporting burden for this collection of information is estimated to average 1 hour per response Send comments regarding this burden estimate or any other aspect of this collection of information to Washington headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302, and to the Office of Management and Budget, Paperwork Reduction Project (0704-0188) Washington DC 20503.

1. AGENCY USE ONLY *(Leave blank)*	2. REPORT DATE March 2002	3. REPORT TYPE AND DATES COVERED Ph.D. Dissertation

4. TITLE AND SUBTITLE: A Software Technology Transition, Entropy Based Engineering Model	5. FUNDING NUMBERS
6. AUTHOR(S) Michael S. Saboe	

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) Naval Postgraduate School Monterey, CA 93943-5000	8.PERFORMING ORGANIZATION REPORT NUMBER NPS-SW-2002-02

9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES) US Army Next Generation Software Engineering Technology Area US Army Tank Automotive Research, Dev. and Eng'g Center, Warren MI , 48397-5000	10. SPONSORING / MONITORING AGENCY REPORT NUMBER NEXTGEN-2002-3-001

11. SUPPLEMENTARY NOTES The views expressed in this thesis are those of the author and do not reflect the official policy or position of the Department of Defense or the U.S. Government.

12a. DISTRIBUTION / AVAILABILITY STATEMENT Approved for public release; distribution is unlimited	12b. DISTRIBUTION CODE

ABSTRACT Currently, systematic techniques for assessing macro mechanisms for transferring software engineering technologies are non-existent. This leads to inefficient allocation of research resources and increased risk to software technology intensive programs. Consequently, software technology transition today is an ill-defined and non-repeatable, inefficient process for bringing advanced software engineering technologies to market. This dissertation develops the fundamental elements of an *industrial model* of a *software technology transition engine*. The mechanisms are developed utilizing information theory, communication theory, chaos control theory, and learning curve principles. The combination of those scientifically sound mechanisms provides a basis for assessing, and / or prescribing a portfolio of technologies and the implementing macro infrastructure. This provides the *theoretical framework for a practical method for a program manager to establish a high capacity transition channel, which accelerates technology maturation and insertion*. Data samples assess the following technologies: software engineering, software technology transfer, Ada, Java, abstract data types, rate monotonic analysis, cost models, software standards, and software work breakdown structures. An extensive annotated bibliography on software technology transfer and related references, and a bibliography including related material from philosophy, psychology, math, physics, thermodynamics, management, economics, game theory, technology transfer, software engineering, and systems engineering is included.

SUBJECT TERMS Software Engineering, Technology Transfer, Information Theory, Communication Theory, Statistical Mechanics, Dynamical Systems, Control Theory, Learning Curves, Entropy, Information Temperature, Temperature of Software (^{o} Saboe), Technology Transfer Dynamics, Research Management, Diffusion of Innovation, Project Management, Physics of Software	15. NUMBER OF PAGES 526
	16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT Unclassified	18. SECURITY CLASSIFICATION OF THIS PAGE Unclassified	19. SECURITY CLASSIFICATION OF ABSTRACT Unclassified	20. LIMITATION OF ABSTRACT UL

THIS PAGE INTENTIONALLY LEFT BLANK

**A SOFTWARE TECHNOLOGY TRANSITION ENTROPY BASED
ENGINEERING MODEL**

Michael S. Saboe
GS-15, U.S. Army
B.S.M.E., University of Bridgeport, 1976

Submitted in partial fulfillment of the
requirements for the degree of

DOCTOR OF PHILOSOPHY IN SOFTWARE ENGINEERING

from the

**NAVAL POSTGRADUATE SCHOOL
March 2002**

Author: _____
                                   Michael S. Saboe

Approved by:

_____	_____
Valdis Berzins	Luqi
Professor of Software Engineering	Dissertation Committee Chair
Software Engineering	and Supervisor
_____	_____
Barry Boehm	Daniel Boger
Professor of Computer Science	Professor of Computer Science
University of Southern California	
_____	_____
Vladan Jovanovic	John Osmundson
Professor of Computer Science	Professor of Engineering
University of Southern Georgia	

Approved by: _____
                         Luqi, Chair, Department of Software Engineering

Approved by: _____
                         Carson K. Eoyang, Associate Provost for Academic Affairs

THIS PAGE INTENTIONALLY LEFT BLANK

ABSTRACT

Currently systematic techniques for assessing macro mechanisms for transferring software engineering technologies are non-existent. This leads to inefficient allocation of research resources and increased risk to software technology intensive programs. Consequently, software technology transition today is an ill-defined, non-repeatable, and inefficient process for bringing advanced software engineering technologies to market.

The essence of this research is defining an engineering model for an evolving software process. The contribution can be summarized as developing the relationships of information "temperature" (^{o}*Saboe*), entropy, pressure, volume (nodes) and the conserved property – information in terms of messages. This ties together for the first time, information theory, chaos control dynamical systems, statistical mechanics and software engineering.

This dissertation develops an engineering model and the relationships of various controlling parameters in an evolutionary process. Cast in terms of new technology transfer (*TechTx*) models for analysis, it is able predict and prescribe action for a research or program manager. Each model deals with entropy as defined in information theory. Each model deals with entropy as defined in information theory. The *TechTx Basic Entropy* model developed addresses macro level trends of a technology at the community level. The *TechTx Entropy Feedback* model is based on non-linear control theory.

The controlling parameter of the evolutionary process is suggested to be information temperature (^{o} *Saboe*), which is developed four different ways. First by comparing the slopes of the controlled property (information in terms of messages). Second, using a one-dimensional set of non-linear dynamical system of equations, then with a two-dimensional system of equations. Third, by using the partition function. With the partition function, the conserved property is allocated to sets of sets in a power set. A probability distribution is developed for discrete message levels, called "q-levels". Each discrete "q-level", which indicate whether there are single terms in a set (q-level=1), a set of sets consisting of pairs of terms is considered q-level=2. q-level=3 consists of a set so sets comprised of three terms, etc. contains a count of the micro-states of primitive messages in that partition. A relationship to the Weibiull distribution function is shown.
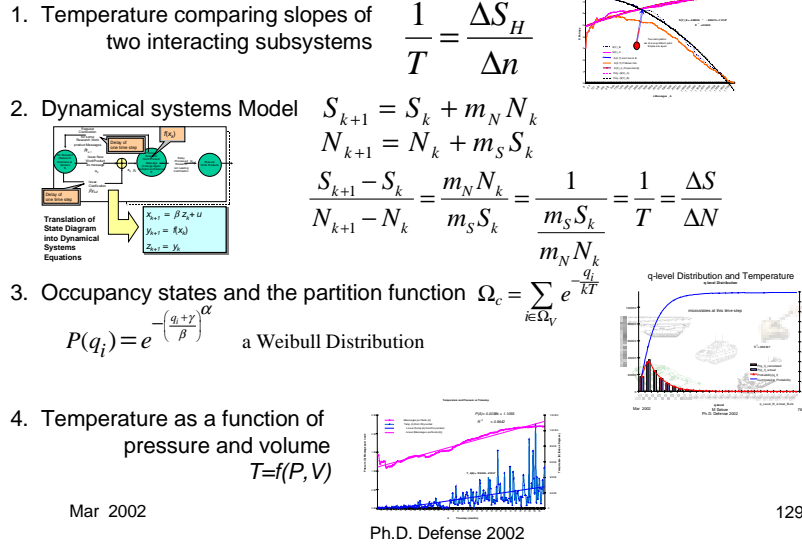
Temperature of Software ($^oSaboe_s$)
four ways

1. Temperature comparing slopes of two interacting subsystems
$$\frac{1}{T} = \frac{\Delta S_H}{\Delta n}$$

2. Dynamical systems Model
$$S_{k+1} = S_k + m_N N_k$$
$$N_{k+1} = N_k + m_S S_k$$
$$\frac{S_{k+1} - S_k}{N_{k+1} - N_k} = \frac{m_N N_k}{m_S S_k} = \frac{1}{\dfrac{m_S S_k}{m_N N_k}} = \frac{1}{T} = \frac{\Delta S}{\Delta N}$$

Translation of State Diagram into Dynamical Systems Equations
$$x_{k+1} = \beta z_k + u$$
$$y_{k+1} = f(x_k)$$
$$z_{k+1} = y_k$$

3. Occupancy states and the partition function $\Omega_c = \sum_{i \in \Omega_V} e^{-\frac{q_i}{kT}}$
$$P(q_i) = e^{-\left(\frac{q_i + \gamma}{\beta}\right)^{\alpha}}$$
a Weibull Distribution

q-level Distribution and Temperature

4. Temperature as a function of pressure and volume
$$T = f(P, V)$$

Mar 2002

Ph.D. Defense 2002

129

Figure i   Four Views of the Controlling Parameter – Information Temperature

The q-level primitive message micro states, empirical data, is related to the partition function, which is found to have a temperature term as the controlling parameter. This result is due to the normalizing condition being primitive message per unit volume. A unit volume in the control space is a performing node. Empirical data for Ada and Java show that the information temperature is similar to that of the ideal gas law. Temperature is proportional to pressure, which can be found to be messages per node.

It is suggested that "the fundamental" units of temperature are in information units.

A most interesting development is the relationship that appears to exist between the two dimensional system of non-linear dynamical equations representing deterministic chaos and the general form of the bakers transformation. The bakers transformation is a general form of a Bernoulli shift, and has been suggested to represent deterministic chaos in evolving processes (Prigogine 1983, 1997). Unlike Prigogine's work, this research suggests for the first time that a system of equations which includes both an abstract representation of a conserved property (information in terms of primitive messages) and

entropy (in information units of bits) have a relationship to a controlling intensive variable – temperature in information units.

The research includes a comprehensive review of the state-of-the-art in software technology transfer. This summary focuses on the elements of technology transfer required to model the technology transfer process. Specifically, this research develops the fundamentals for a rigorous software technology transfer model as required by the *TechTx Entropy Feedback* model. The relationship of entropy (S_H) as defined for information by Shannon, and the eigenvalue, or the norm of a dynamical system, is explored. The Lyapunov number is a natural measure developed from the eigenvalue of a dynamical system, e.g. related to entropy. The significance of the eigenvalue for a communications software technology transfer model is discussed. The result of this research is the definition of an engineering model for an evolving software process.

The mechanisms are developed utilizing information theory, communication theory, chaos control theory, statistical mechanics, and learning curve principles. The combination of those scientifically sound mechanisms provides a basis for assessing, and/or prescribing a portfolio of technologies and the implementing macro infrastructure. This provides the *theoretical framework for a practical method for a program manager to establish a high capacity transition channel, which can accelerate technology maturation and insertion*. The significance of the eigenvalue of the dynamical system is discussed and related to the Lyapunov exponent and number to indicate stability. The relationship to *pressure* on the community, and a *temperature* of the technology process is developed. An engineering model results using a state equation similar to that used by engineers to define a process cycle. The result is useful to program managers, policy makers and practitioners in analyzing and prescribing a process for the evolution of a technology. It is speculated that the state relationships of the Technology Dynamics model can be used to model any evolutionary process and software itself is a special case of the model. Finally, it is suggested that this is the engineering and mathematical basis for *software physics*. Data samples assess the following technologies: software engineering, software technology transfer, Ada, Java, abstract data types, rate monotonic analysis, cost models, software standards, and software work breakdown structures. Also

included is an extensive annotated bibliography on software technology transfer and related references, and a bibliography including related material from philosophy, psychology, math, physics, thermodynamics, management, economics, game theory, technology transfer, software engineering, and systems engineering.

Let's set a context.

Induction is a process of inferring a general law or principle from the observations of particular instances. This is inductive inference. Inductive reasoning is a more general concept than inductive inference. It is a process of assigning a probability (or credibility) to a law or proposition from observation of particular instances. Inductive inference draws conclusions on rejecting or accepting a proposition, possibly with out total justification. Inductive reasoning only changes the degree of our belief in proposition. Deductive reasoning of inference derives the absolute truth or false hood of a proposition. This is a case of inductive reasoning. This approach to explaining things around us dates back at least to Epicurus (342?-270?BC) (Li 1993, p. 274). Let's consider theory formulation in science as the process of obtaining a compact description of past observations together with future ones.

Let us suggest that the preliminary data of an investigator, the hypothesis proposed, the experimental design and setups, the trials performed, the outcomes obtained, the new hypothesis formulated, etc, can be encoded as an initial segment of an infinite binary sequence. The investigator obtains increasingly longer initial segments of an infinite binary sequence by performing more and more experiments. To describe the underlying regularity in the sequence, the investigator tries to formulate a theory that governs the sequence on the basis of the outcome of past experiments. Candidate theories or hypothesis are identified from the sequences starting with the observation of the initial segment.

There are many different possible infinite sequences or histories that the investigator can embark on. The phenomenon the investigator is trying to understand or the strategy used can be stochastic. In this type of view, a phenomenon can be identified with a measure, i.e. probability distribution, on a continuous sample space.

This research attempts to express the task of learning a certain concept as in terms of sequences over a basic alphabet. We express what we know as a finite sequence over the alphabet, an experiment to acquire more knowledge is encoded as a sequence over the alphabet, the outcome is encoded over the alphabet, new experiments are encoded over the alphabet and so on. This way we can view a concept as a probability distribution (measure) over a sample space of all one way infinite binary sequences. Each sequence corresponds to one never ending sequential history of conjectures, refutations, and confirmations. The distribution can be said to be the concept of phenomenon involved. We can predict what is likely to turn up next with an initial segment. Using Bayes rule, for conditional probability, we can predict and extrapolate future outcomes. This is the general thrust of this research.

Hope you find this interesting. There is a lot more here than meets the eye.

THIS PAGE INTENTIONALLY LEFT BLANK

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

ACKNOWLEDGMENTS

*Hafiz, whose given name was Shams-ud-din-Muhammad, is the most beloved poet of Persia. Born in Shiraz, he lived at about the time of Chaucer in England and about a hundred years after Rumi. He spent nearly all his life in Shiraz, where he became a famous Sufi master. When he died, he was thought to have written an estimated 5000 poems of which 500 to 700 have survived. His **Divan** (collected poems) is a classic in the literature of Sufism. The work of Hafiz became known in the West largely through the efforts of Goethe, whose enthusiasm rubbed off on Ralph Waldo Emerson, who translated Hafiz in the nineteenth century. Hafiz's poems were also admired by such diverse writers as: Nieztsche, Pushkin, Turgenev, and Garcia Lorca; even Sherlock Holmes quotes Hafiz in one of the stories by Arthur Conan Doyle. In 1923, Hazrat Inayat Khan, the Indian teacher often credited with bringing Sufism to the West, proclaimed that "the words of Hafiz have won every heart that listens."*

Hafiz's poetry is rooted in the beautiful human need for companionship and in the soul's innate desire to surrender all experience -- except Light. The verses speak on many levels simultaneously, though they are crafted with such brilliance, rarely does one feel left out.

People from many religious traditions share the belief that there are always living persons who are one with God. These rare souls disseminate light upon the earth and entrust the Divine to others.   Hafiz is regarded as one who came to live in that Sacred union, and sometimes his poems speak directly to that experience.

If God wanted, He could give Himself entirely to someone without diminishing His own state. And if you were the recipient of that Divine Gift -- what would you then know?

Rumi, Kabir, Saadi, Shams, Fransis of Assisi, Ramakrishna, Nanak, Milarepa, and Lao-tzu are among the many known to have achieved perfection or Union because their extraordinary romance with the Beloved. They are sometimes called the "realized souls" or "Perfect Masters." My Father, Michael S. Saboe Sr., was one of those. He, as have my other mentors and advisors, have greatly enabled the work I have done.

As Hafiz wrote:

**The voice of the river that has emptied into the Ocean
Now laughs and sings just like God.**

I. INTRODUCTION

A. GOALS AND PROPOSED NEW CONTRIBUTION

1. The Problem and Goals

Software Technology transition today has an ill-defined and non-repeatable, inefficient process for bringing advanced software engineering technologies to market.

> *The goal of this research is to develop the basic elements for an industrial model of a software technology transition engine that establishes a high capacity transition channel, which accelerates technology maturation and insertion.*

The top level requirement of the model is to minimize the amount of effort required to realize an idea into reality. A set of concepts is introduced that are cycle, application and technology independent. This research presents a general set of models, with underlying independent and dependent variable relationships for software technology transition. The model is an engineering model in the full sense. The underlying model is as robust as any thermodynamic or physics model. It represents a closed form of interrelated equations that are brought to the software engineering discipline for the first time. These models provide a method to analyze and later prescribe the size of a research transition infrastructure and the probability of a technology maturing at a given time. Further, the engineering and mathematical relationships appear to be applicable to any evolutionary process e.g. software development) and potentially to software itself.

This research dissertation develops the elements of three new technology transfer models that can be represented mathematically. This provides a method for analysis for both predictive and prescriptive activities. All of the existing work in software technology transfer appears to lack mathematical models. The three technology transfer

models addressed are: 1) *TechTx Basic Entropy*, 2) *TechTx Entropy Feedback*, and a 3) *TechTx Entropy Learning Curve* is suggested.

The basic model analyzes the entropy[1] of terms relating to technology messages published[2] over time. This model is compared to a baseline model, a message vs time, used in the diffusion of information research literature. The second model is at the organizational node or sub-node level and gives the basis for analyzing macroscopic and local interactions in a process. The third model suggests the incorporation of learning curves at the organizational node level. This model is refined to incorporate both entropy and learning. Each of these models represents a refinement of the predecessor model. For example, 3) TechTx Entropy Learning Curve, builds from 2) TechTx Entropy Feedback, which is an extension of both 1) TechTx Basic Entropy. The mathematical implications of the third model are suggested. While all three models represent an extension to the state-of-the-art, the last model, TechTx Entropy Feedback, provides the basis for an entire set of engineering tools to permit analysis of a evolving processes. This model is validated and the results of over 100,000 data points yield a confidence interval of less than ±0.3% .

The key underlying communication diffusion research of Rogers (Rogers 1983, 1995) is pervasive in the more specific study of software technology transfer (see Buxton 1991, Raghavan 1988, 1989, Fichman 1993, 1994, Jaakkola 1995, Fowler 1994, Pfleeger 1999, and many more). The research in this dissertation suggests preliminary analysis of the basic elemental tools required for a software technology transition cycle analysis approach.

This work is motivated (see B.　　Motivation and Significance of the Problem, p4) by the need for an acquirer, or research program manager, to assess risk related to the maturity date of a technology. Data and charts that summarize relevant aspects of this

---

[1] Entropy (in greek it comes from *en*, in + *trope*, turning) comes from the conviction that the future will not repeat the past, that time moves unidirectionally, and the world is moving on (Nash 1974). By always increasing in the direction of spontaneous change, entropy indicates the "turn," or direction, taken by all such change.

[2] Unless otherwise noted: the words "publish", and "sent" are used interchangeably; the words "message", "publication", and "record" are synonymous. Publishing a message is the same as performing a task to develop a message.

work are presented. A sample data set for "software engineering" is plotted. "Technology Transition Models" (see A. technology Transfer Model Features, p31) then summarizes the specific software technology transfer literature. Most of this literature addresses the implementation details required to address software technology transition.

With the principle relationships of the models developed, the research suggests methods to construct and analyze a design for a technology transfer engine. The design can provide prescriptive insight to a program manager or research manager, as to how to best configure a research program to achieve stability, confidence, and earliest convergence.

B. MOTIVATION AND SIGNIFICANCE OF THE PROBLEM

At the *International Conference on Software Engineering 2001,* the keynote speech (Shaw 2001) illustrated the trends in maturation of software technology. The model cited was one from 1984 (Redwine 1984). That model, while the result of an interesting set of case studies at that time, provides no prediction capability. It only identifies a set of state transition points, to tag an historical analysis in other case studies. Two of the states identified in that model are not identifiable in any consistent manner, and have been questioned in the literature (Pfleeger 1999), (Saboe 2001).

Current military applications typically push high performance technology without large consideration given to cost. On the other hand, commercial enterprise applications are very much interested in producing a product with reduced cost, increased responsiveness to market pressures, and reduced cycle time to product delivery.

The current model in use in the United States features the National Science Foundation (NSF) and Department of Defense (DoD) as major contributors to the advancement of software technology (e.g. NSF, Defense Advanced Research Projects Agency and Service Laboratories). There has not been a focused national implementation effort in the high technology area of software engineering, although it has become a national agenda item (Boehm, Basili 2000).

The approach to date has been criticized for decades in numerous government reports and in the literature (DSB 2000). The current approach, to advance software-engineering technology, is a by-product of some advanced technology development effort that focuses a narrow light on the requirements of the target system. The large ticket NSF, DARPA, and Service lab efforts in software engineering tend to move in parallel to advanced system developments. Historically, these efforts are always looking for a home and an insertion point. Yet, the product developers desire mature technologies that work well in the field, not whiz-bang lab tools that work fine only in the fabricated demonstrations. This poses a problem for efficient, consistent insertion. It also highlights a waste of national intellectual capital.

In the commercial model, let's simply look at the challenges of Microsoft and Netscape. The competitive challenges as well as the challenges of immature technologies that are rapidly emerging as standards (Cusumano 1995 and 1998) brightly illustrate the obvious. Industry needs a better model for inserting technology as well. Another development is the general movement to standards-based software (Jovanovic 1999). Not only are we moving towards open standards and infrastructure in software applications, but also in vehicles with embedded software, and in the software engineering organization. It turns out that the weak area in the Redwine model is exactly in the area related to diffusion, or popularization to the broader population. This popularization phase is the point where the standardization phase of a technology occurs.

C.   DESIRABLE FEATURES AND DEFICIENCIES

It seems reasonable to define some desirable characteristics of a good software engineering technology transition model. The model should enable the software research and program manager community to quantify the maturation of a technology (or portfolio of technologies), and the uncertainty in the arrival time of the technology. With the appropriate analytical model, we should be able to manipulate the model to enable adjustments and prescriptions.   Primary reason to analyze, adjust and prescribe is to reason about ways to reduce relative risk and uncertainty, and accelerate the arrival of a technology for use in a program.

After a careful review of the literature, it seems apparent that a good model for technology maturation and transition is lacking for software engineering. There are no references in the software technology transition literature, which indicate that a model for analyzing, predicting, and prescribing maturation, stability, and confidence in the evolution of a technology exists.   There is a clear need based on the researcher's extensive personal experience (nearly 30 years at every level of industry and the Department of Defense). Discussions with the software technology transition program at the Software Engineering Institute (SEI), consistently indicated that there is a critical lack of and need for an analytical model of the type proposed.   The elements of such a proposed analytical model promise to permit analysis of various alternatives for policy and investment trades.   Tools that build on this analysis approach can help identify leverage points and opportunities to accelerate progress in a repeatable and rigorous process enabling quantification of maturity at a given date and confidence in a subject technologies stability.

With such tools, a decision-maker can determine the confidence with which a technology or group of technologies will stabilize and converge in a given time frame. For example (see Figure I-1), a risk assessment use for a program might expect a portfolio of technologies to arrive by year 06 with an 80% certainty, but the model might show that in 06, there is only 60% certainty of being available using the current trends. The desired 80% certainty would not be available until 08. For the desired system, or macroscopic, curve, we can algebraically solve for the node response curve(s).   The

model can then be used for prescriptive purposes. This enables trades to determine how many and whether parallel or serial tasks are required. If the technology is not predicted to arrive as required, the model will point to the areas for remedy with a prescriptive solution to *organize, train and equip* an organization in order to change the confidence of arrival of the technology for the program's required schedule.

Program Office Use for Risk Assessment and Rx

Example:
Program Office Wants
by 06 with 80% certainty

Analysis indicates 08

What nodes / programmatics
need to be put into place to
shift curve to left?

From desired system curve
Algebraically solve for node response curves(s)

Determine how many and parallel / serial
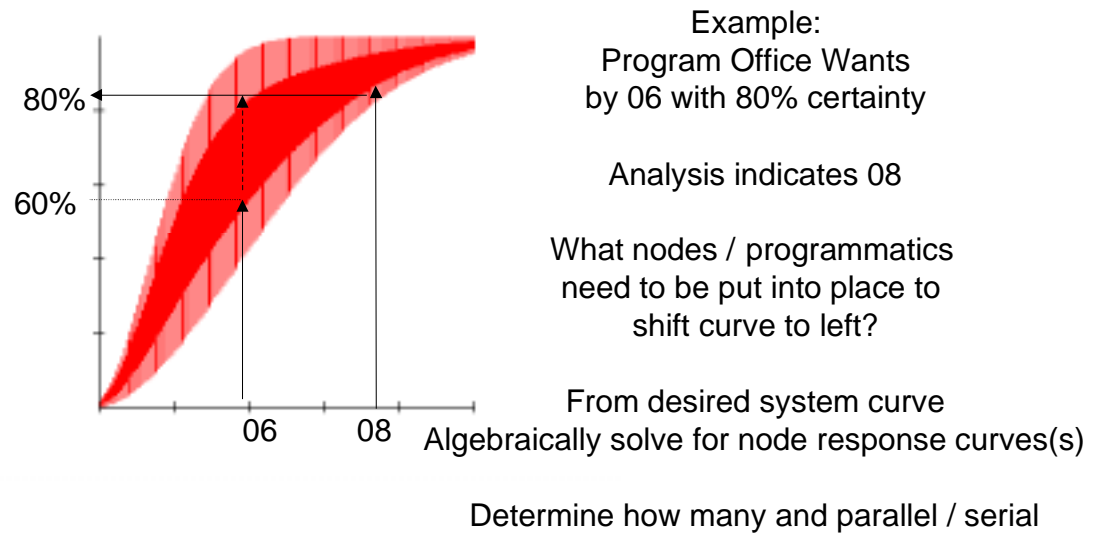
Figure I-1 Program Office Use of Objective Model

D. RESEARCH APPROACH

1. Rational and Experiential Analysis

The study of abstract thought has persisted and evolved along with the emergence of experimentalism. A well-known marker in the intellectual history of this study is Rene Descartes. Historically, it has been thought that Descartes established the proper method of inquiry with his statement *Cogito, ergo sum*, "I think, therefore I am"[3]. Roger Bacon (Haskins 1927) later, Newton[4], Lock, Barkley[5] and others brought us to the sensible experiential flavors. We review the development of this merging of philosophy, math, physics, and metaphysics with the practical experimental methods we use as engineers.

As engineers, we assimilate, combine and produce. Good technology is contrived to fulfill a human need. That is why it satisfies more than function. This research assumes, as a basic premise, that software technology transfer is not significantly different from the development of knowledge in other disciplines. The subject matter, or domain, is different, but the constructs used by humans to formulate physical or experimental knowledge are similar. The game, then, is to meld the logical-mathematical philosophical musings represented in a model with information gathered to validate the model in order to reduce uncertainty, and to communicate the results. Thus, we have the intention of diffusing the information to the society or subsets of the society (receptors) that use the information gathered in the development or extension of a technology, and to

---

[3] It is often misunderstood that this statement represented the "proof" of his existence, vice the method of rational analysis and an examination approach devoid of the defects of perceptions. Even with rigorous experimentation our "perceptual" and "sensual" observations of associations of properties are often fooled. This discussion turns up repeatedly through the history of science. Even the defeat of pure skepticism occurs due to uncertainty. It is a curious aside to note that it was not until the early twentieth century that the scientific method evolved to the point of rejecting the null hypothesis.

[4] There are two linkages here, the 1st law and state. Newton, when formulating his laws was improving on Descartes' *Principia.* Newton learned about the law of inertia from Descartes. In fact, it is the first law in both the *Principia* of Descartes, and the *Principia* of Newton, and both deal with "continuous" acting forces. From Descartes' presentation of the law, Newton learned the important concept of motion as a "state" (status) (Newton 1726, p46). He developed the 2nd law, which sets forth a proportionality between a "force" and a "change of motion." In this law, it means an impulse, a discrete force. The 1st law was formulated (as a hypothesis) to allow for the condition that there were certain [continuos] insensible forces that are otherwise not known to use (Newton 1726 p110). We could speculate that there could be a counterpart today for discrete "forces" not otherwise know to us – say an information force.

[5] Barkley gives us the saying that goes like this, if a tree falls in the woods and no one hears it does it make a noise? A message is communicated only if there is a receiver to receive it.

the subsets (consumers) that would use a technology. The research in this dissertation, in a limited sense, is studying that process itself.

2.   Context and Overview

Let's set a context. Induction[6] is a process of inferring a general law or principle from the observations of particular instances. This is inductive inference. Inductive reasoning is a more general concept than inductive inference. It is a process of assigning a probability (or credibility) to a law or proposition from observation of particular instances. Inductive inference draws conclusions on rejecting or accepting a proposition, possibly without total justification. Inductive reasoning only changes the degree of our belief in proposition. Deductive reasoning of inference derives the absolute truth or false hood of a proposition. This is a case of inductive reasoning.

This approach to explaining things around us dates back at least to Epicurus (342?-270?BC) (Li 1993, p. 274). Let's consider theory formulation in science as the process of obtaining a compact description of past observations together with future ones. Let us suggest that the preliminary data of an investigator, the hypothesis proposed, the experimental design and setups, the trials performed, the outcomes obtained, the new hypothesis formulated, etc., can be encoded as an initial segment of an infinite binary sequence. The investigator obtains increasingly longer initial segments of an infinite binary sequence by performing more and more experiments. To describe the underlying regularity in the sequence, the investigator tries to formulate a theory that governs the sequence based on the outcome of past experiments. Candidate theories or hypotheses are identified from the sequences starting with the observation of the initial segment.

There are many different possible infinite sequences or histories on which the investigator can embark. The phenomenon the investigator is trying to understand or the strategy used can be stochastic. In this type of view, a phenomenon can be identified with a measure, i.e. probability distribution, on a continuous sample space.

---

[6] The Oxford English Dictionary defines induction this way.

This research attempts to express the task of learning a certain concept in terms of sequences over a basic alphabet. We express what we know as a finite sequence over the alphabet. An experiment to acquire more knowledge is encoded as a sequence over the alphabet, the outcome is encoded over the alphabet, new experiments are encoded over the alphabet, and so on. This way we can view a concept as a probability distribution (measure) over a sample space of all one way infinite binary sequences. Each sequence corresponds to one never ending sequential history of conjectures, refutations, and confirmations. The distribution can be said to be the concept of phenomenon involved. We can predict what is likely to turn up next with an initial segment. Using Bayesian analysis (Bayes 1763) to compute the conditional probability, we can predict and extrapolate future outcomes. This is the general thrust of this research.

Let's develop an analogy of the flow of communication to a physical model to illustrate the concept. When two people meet, they converse, and consequently modify their thinking to some extent. These modifications are brought to subsequent meetings and modified further. The word for this is dissemination or diffusion. There is a flow of communication in society, just as there is a flow of correlations in matter. Let's explore this idea of correlations using the analogy of a physical system and look at what happens in terms of distribution functions.

Consider a glass of water. We may visualize the interactions as leading to collisions between the molecules. We can describe the water containing them in terms of a statistical ensemble. The water is not aging if we were to consider the individual molecules over geologic time[7]. Yet, there is a natural time order in the system from a statistical point of view. Aging is a property of populations, as in the biological theory of evolution as developed by Darwin. It is a statistical distribution that approaches the equilibrium distribution.

[7] Newton's scholium differentiates time this way. "Time, space, place, and motion … quantities are popularly conceived solely with reference to the objects of sense perception. … 1. Absolute, true, mathematical time, in and of itself and of its own nature, with out reference to anything external flows uniformly and by another name it is called duration. Relative, apparent, and common time is any sensible and external measure (precise or imprecise) of duration by means of motion; such a measure – for example, a month, a year – is commonly used instead of true time." (Newton 1726 p408). This annotated translation keeps to Newton's original language. Many translations have been modernized. These other modernizations do not lend itself to the rich abstract nature, and subtulies that are important to this research.

Consider a probability distribution $p(x_1, x_2)$ dependent on the two variables x_1, x_2. If x_1 and x_2 are independent, we can factor $p(x_1, x_2) = p_1(x_1) p_2(x_2)$. The probability $p(x_1, x_2)$ is the product of the two probabilities. On the other hand, if $p(x_1, x_2)$ cannot be factored, x_1 and x_2 are correlated (Bayes 1763 p299) Return to the glass of water molecules. The collisions between the molecules have two effects: they make the velocity distribution more symmetrical, and they produce correlations (see Figure I-2). However, two correlated particles will eventually collide with a third one (see Figure I-3). Binary correlations are then transformed into tertiary ones etc. Prigogine illustrated this molecular model, and it has been verified (Prigogine 1997 p79).

Collisions and Correlations

After Collision

The coll...es creates a correlation between them
...ted by the wavy line)

Source: After Prigogine 1997

Nov 2001   M Saboe   71
                      Ph.D. Defense 2001
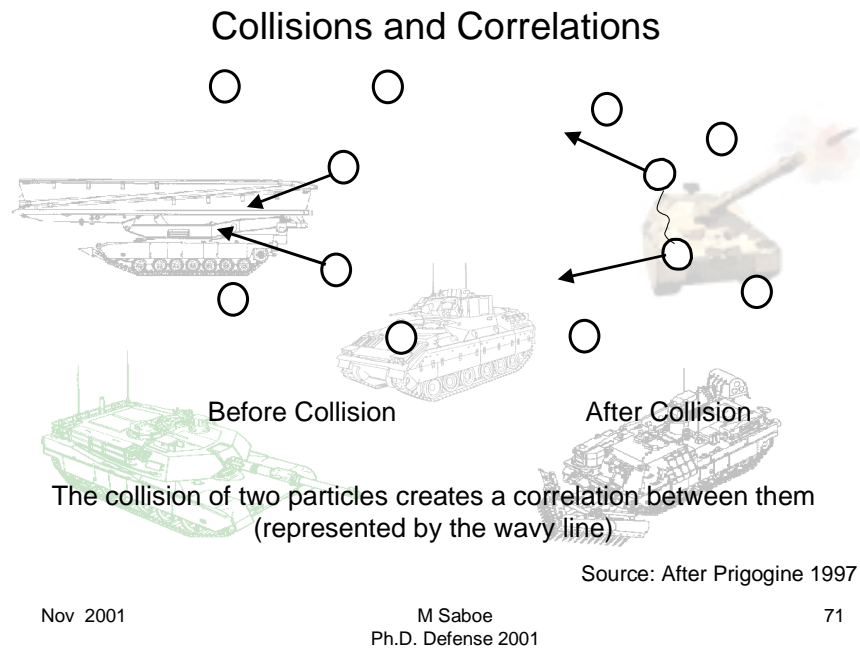
Figure I-2 Collisions and Correlations (Source: After Prigogine 1997)

Flow of Correlations

Before Collision   After Collision

Successive collisions lead to binary, tertiary, ... correlations

Source: After Prigogine 1997

Nov 2001   M Saboe   72
                     Ph.D. Defense 2001
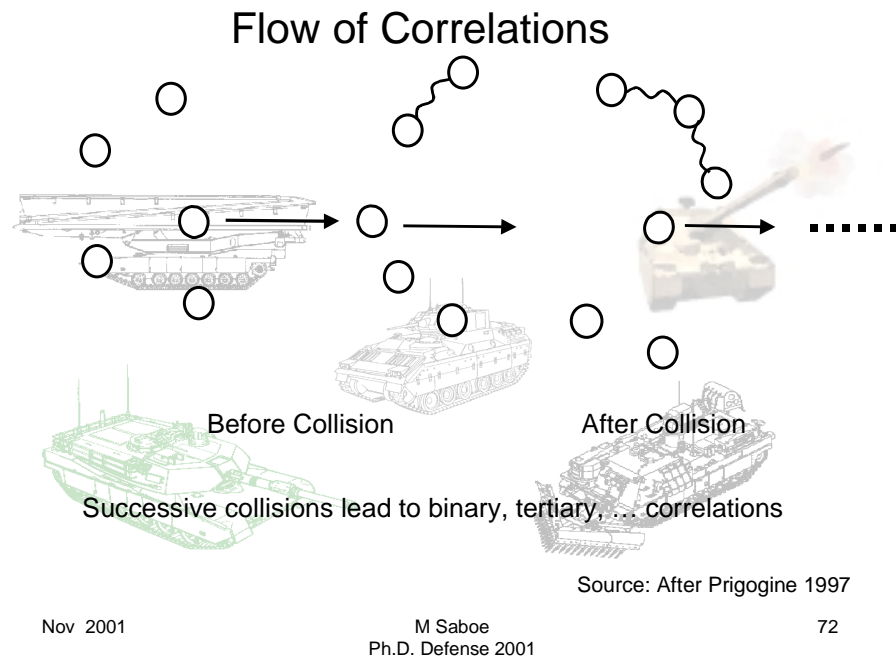
Figure I-3 Flow of Correlations (Source: After Prigogine 1997)

We could conceive of inverse processes that make the velocity distribution less symmetrical by destroying correlations. Processes that invert the velocity of particles for a physical world as in Figure I-4 have been reproduced. However, this inverted flow of correlations can only be achieved for a short time, with limited numbers of particles. Then we again have a directed flow of correlations involving an ever-increasing number of particles leading the system to equilibrium.

We now have a flow of correlations that are ordered in time just as there is a flow of communication in society. There is a method to describe this irreversibility. This statistical description is dynamics of correlations leading to the equilibrium solution.

In this research, we use messages instead of particles. This turns out to be a conserved quantity (conserved quantities shared between two systems need not be restricted to energy[8], or mass, or volume, the conserved quantity could be a number of

---

[8] Energy is an interesting term. It is a primitive term. It is a mathematical abstraction that has no existence apart from its functional relationship to variables or coordinates that do have a physical interpretation and that can be measured (Abbott 1989 p1). The 1st law of thermodynamics is merely a

measures, even money) (Yakavenko 2000) (Farmer 1999). We are concerned with a deterministic dynamical system as well as an especially simple type of dynamical system, both corresponding to dynamical system maps. Contrary to what occurs in ordinary dynamics, time in maps acts only at discrete intervals. Maps represent a simplified form of dynamics that make it easy to compare the individual level of descriptions (trajectories) with the statistical description (see Appendix A Information, Control Theory and Evolutionary Dynamical Systems Basics, p273). (Prigogine 1997 p81).

It is not the place of this research to provide a mathematical formalism with theorems and lemmas. Rather this research provides a heuristic solution. We do, however, want to recognize that the careful construction of the model aligns with very deep mathematical constructs. It is important to realize that the problem of correlations of information distributions and dynamical systems can not be solved at the level of trajectories or individual particles. It can, however, be solved at the level of ensembles[9]. In the TechTx Entropy Learning Curve Model, the sample space is allocated to course grained partitions. In this way, we can connect the dynamical and statistical views in a manner that is consistent with the newest chapters in math and physics. We are able to predict the speed at which the distribution approaches equilibrium and to establish the relationship of this speed with the Lyapunov exponent[10]. This is developed in Chapter III.

---

formal statement asserting that energy is conserved. This represents a primitive statement about a primitive concept. Moreover, both are linked. The 1st law depends on the concept of energy, and it is equally true that energy is the *essential* thermodynamic function precisely because it allows the formulation of the 1st law.

[9] Boltzman was the first to show the relationship of trajectories in state space and ensembles. It is his work that is considered the first practical use of statistical mechanics.

[10] The Lyapunov exponent shows a divergence or convergence in dynamical systems. This identifies the signature of a deterministic dynamical system.

Destruction of Correlations

a) Particles (black points) interact with obstacle (circle) Initially all of the particle... velocity. The collis... ...locities and create... ...een the part... ...e

b) represents the opposite process. Consider the effect of velocity inversion as the result of the inverted collision. Correlations with the obstacle are destroyed, and the initial conditions are recovered.

Source: After Prigogine 1997

Nov 2001   M Saboe   73
                    Ph.D. Defense 2001

Figure I-4 Destruction of Correlations

3.   Validation

The research validation follows the strategy shown in Figure I-5. The proposed *TechTx Basic Entropy* model asks the question, "X is a method of predicting technology maturity, Can we do better?" in assessing the maturity of a technology, using the Y, new model. Validation compares it to the existing methods. Constructing the *TechTx Entropy Feedback* model is a more difficult challenge. Development was difficult due to the lack of previous work in the software community in relating statistical mechanics, non-linear dynamical systems control theory and information theory. Validation proved straight forward, since the model lent itself to readily collecting samples to validate the equations with thousands, to over one hundred thousand data points. Here the research is asking, "Can it be done at all?" The *TechTx Entropy Feedback* model was developed and exceeded expectations. The model is exercised with data from the *TechTx Basic Entropy* model. The *TechTx Entropy Learning Curve* model is suggested from the results of the other models. The technology transfer maturation process is characterized by learning curves. The validation is of the form, that Shaw used, "Look, it works!!" (Shaw 2001)

Question   **Strategy/Result**   **Validation**

TechTx Feedback Model

| **Can X be done at all?** | Qualitative model | **"Look, it works!!"** |

TechTx Entropy Learning Curve

| Characterization | Technique | Implementation |

TechTx Basic Entropy

Method/Means	System	Evaluation
Can X be done better?	**Build a Y that does X**	**Measure Y, compare to X**
Generalization	Empirical model	Analysis
Selection	Analytic model	Experience

Figure I-5 Validation Strategy (Source: After Shaw 2001)

The proposed model was compared with the traditional diffusion of innovations communication model to predict trends and the maturation of a technology. The traditional model is the baseline model and uses the *message-counting* method. The first proposed model is the *TechTx Basic Entropy* model. This is the first improvement over the traditional model and uses the content of the message, measured in the information dimension of entropy. The entropy is derived from the basic message counting model so the excellent predictions seen by the linear message counting model is retained. Historically, entropy is represented in information units - bits. Essential elements related to entropy are addressed in Chapter II. These include, 1.   Probability, 2. Information, Uncertainty, 6.Stochastic Model and Markov Chains, and related concepts (see C.

Statistical Elements of the Technology Transition Models, p69). Chapter III, (see

1.   Entropy Review, p98) includes a brief review of entropy as used in information theory.

The *TechTx Entropy Feedback* model was compared with experimental data to validate the state equation relationships, information theory and dynamical systems equations.

The last model is the *TechTx Entropy Learning Curve* model. It appears that the feedback model exhibits characteristics of learning curves. With the addition of the learning curve to the feedback model, this model suggests a method which determines the learning rates for organizations and researchers (on average) in performance bands of +/- 1σ, +2 σ, +3 σ, and greater than 3 σ. It is an extension of the *TechTx Basic Entropy* model. The *TechTx Entropy Learning Curve* model, is not validated explicitly, however, the feedback model is tuned to equate entropy measured at the macro (system) level with entropy measured at the micro (organizational node) level. The result appears to be a learning curve. Using a transfer function, this tuning creates the relationship between the macro world entropy of the *TechTx Basic Entropy* model, and the micro world entropy of an organization.

E.   OVERVIEW OF THE DISSERTATION STRUCTURE

This research has progressed in a pattern typical of the history of the development of science throughout the ages. We first set an initial context and historical relations in Chapters I and II. The assessment of previous work in Chapter II introduces existing models used in technology transfer, then concentrates on the issue of software technology transfer. At the end of Chapter V, we speculate that the model is general enough to be applied to any technology, and should not be limited to the domain of software. Since the proposed model relies heavily on the concepts related to the learning curve, statistical mechanics and entropy, a review of these concepts is also developed.

A table summarizing the various work and features is mapped to the proposed model contributions. Deficiencies in the current approach to software technology transfer are identified in each section of historical model review. In short, there has not been a systematic, mathematical approach focused on the technology transfer infrastructure. Most work has addressed implementation details.   This effort focuses on the mathematical and logical models of the overall technology transition channel.

We begin with the model development in Chapter III introducing information entropy, and learning curves. The steps include:

1) Development of the macro/micro relationships of information entropy (which are related to statistical mechanics) for software technology transfer.

2) Development of statistical mechanics and dynamical systems relationship to yield technology transfer dynamics models.

The relationship of complexity of an input entropy and number of tasks required to reduce the time per unit task is developed in a stepwise fashion. The approach developed in Chapter III expands the basic linear model with a general form of non-linear components in a dynamical system model. The dynamical system models, first in one dimension (entropy), and then two dimensions (entropy and number of tasks performed) in a time step are developed.   Here we are addressing two orthogonal views of

17

complexity. On one axis, we find information content addressed by information theory, where generally the optimization is around minimizing program length and packing of sequences and patterns. On the other axis, complexity addressed by dynamical systems. Optimization along this axis is generally around minimizing time needed to perform the process. Also this can represent a state space of intensive and extensive variables. This will be discussed in Chapter II, with the review of the Statistical Elements of Technology Transfer, and in Chapter III. Combining these views permits development of a performance index roughly in terms of tasks per unit time to enable trades between program length, and performance. The performance index coupled with the other views in state space provides the mechanism to determine the configuration of a technology transition channel or engine to mature a technology.

A macro view of the system is developed. The macro view is related to the constituent micro (organizational node) level view. Discussion on the tuning of the model parameters of a learning curve at the node level to approximate the true system is developed. A three dimensional extension to the basic models which includes feedback is proposed.

Validation and Assessment of the data in Step 3 is based on data collected on a sample of 50,744 raw records for the eight technologies. For example, a technology with approximately 4250 raw records, comprising an alphabet of 1583 primitive message terms, capable of generating 676,417[11] messages sets – the data points which form the basis of the models. The data was taken in monthly intervals over a 21 year period. The nodes over the same time-period consisted of 22,394 author sets. This gives a very tight confidence interval, which is discussed in Chapter IV. The technologies were chosen because they were assumed to have well studied histories. These technologies include Ada, Java, Abstract Data Types, Rate Monotonic Analysis, Software Cost Models, Software Work Breakdown Structures[12], Software Technology Transfer, and Software

---

[11] The confidence interval can be approximated by $1/\sqrt{n}$. This is $1//\sqrt{117,637} = +/- 0.292\%$ for messages and $1/\sqrt{22,394} = +/- 0.67\%$ for author node sets. Generally, this can be considered a very tight confidence interval.

[12] The author performed significant research in Software Work Breakdown Structures for the Department of Defense in the 1990's. Therefore, it was a technology with a well-known history.

Engineering. The first technology Ada was well studied and like the internet was initially sponsored by a government organization. Java, has a well known history and like Ada there was significant early sponsorship (Sun) but many more users were exposed to this technology over a shorter period of time due to the emerging nature of the world wide web, and standards driven by industry. The next three technologies (Abstract Data Types, Rate Monotonic Analysis, Software Cost Models) were studied elsewhere and offered a set of data for comparison with another model. The remaining technologies were chosen because the subjects were well known to the author, and in the case of software engineering, of general interest to the community. The discussion and validation of the model using these technologies is performed in Chapter IV. A heuristic development approach is used to validate the conclusions. The degree of formality (low) was determined by considering the current maturity of software engineering and its related science, computer science, relative to other disciplines at this stage.

Data is collected on a variety of technologies that have been previously studied. The data is easily collectable and available to decision-makers at the macroscopic, observable, performance parameter level. At this point, the theory development and validation is done. With these models in place, future research can explore cycle analysis and implementation details can be refined.

The appendices provide an overview of relevant advanced mathematical details, general discussion of historical note related to Chapter III, and data used in Chapter IV. The appendix also includes a description of the entropy model codes and data reduction tools developed for this research. The tools used are Microsoft Excel and Access applications. Add-ins, in the form of macros, contains the analytical models. Interface code is written in Visual Basic. While research tools, they are suitable for performing analysis of the type proposed. A significant contribution is the software technology transition annotated bibliography in the appendix. This bibliography provides a data set for future analysis of the feedback model.

Chapter V summarizes the contributions of this research and provides conclusions pointing to the scope of future work. It suggests that analysts are able to develop, from

this point of departure, a point design for an "Industrial Model of a Software Technology Transition Engine".

Implications and future research are identified in Chapter VI. In addition, in Chapter VI, it is suggested that a software technology transition engine could be analyzed with the tools developed. We conjecture that such an engine, one that pumps technologies to the user community, should be efficient, i.e. the maximum amount of work product should get to the goal of insertion with the minimum amount of resources consumed and wasted. An analytical approach is suggested that uses a cycle diagram, familiar to physicists, mechanical engineers and thermodynamicists. The technology transfer TechTx dynamics cycle diagram and analytical approach could be used to evaluate the efficiency of the technology transfer engine. This approach is similar to a Carnot cycle analysis using state[13] points of entropy, temperature, and pressure. Chapter VI suggests areas for additional work: the notion of "squaring the Carnot cycle"; the Second Law Analysis, a description of the TechTx engine in terms of evolutionary software development process; and identification of software development *entropy* metric. Further, since this research has linked its foundation to physics and thermodynamics, we now have the full richness of those disciplines potentially available. This will permit building and extending software engineering with existing theory in these disciplines in the language familiar to the scientist and engineer.

Findings: This research identified a minimum collection of variables that can represent a framework for an industrial strength model for software technology transition. Manipulation of these variables enables analysis of the cause and effect relationship of elements constituting a transition channel. The research suggests a set of relationships that can be manipulated in much the same way that science and engineering disciplines evaluate designs using physics and thermodynamics. The model presentation is suitable to communicate to policy makers. In fact, initial relationships developed in this research suggests that there is a *"software physics"* that can at least be applied to software technology transfer and by extension, to evolutionary software development, and with further research, to the software itself. It may in fact apply to the evaluation of any

---

[13] State comes from the term status.

evolutionary technology system's assessment beyond the discipline of software. This is especially aligned to assist with, biologically inspired computing to compute with patterns, not bits. It appears that this logic development is not obvious if one approaches from the software and traditional deterministic "programming" direction.

F.   DEFINITION OF TERMS

1.   τεχνη (Techne), Science, and Invention

The title of this research pivots around the terms *technology, transition* and *engine*. All of the other terms are simply qualifiers that narrow the domain (software), target the user, and robustness (industrial - implying, albeit loosely, the notion of usage in a non-trivial solution and operational space), and model (implying this product is a representation or approximation).   The terms *high capacity, accelerated,* and *high efficiency* represent desired performance characteristics of the model. There is a desired causal relationship between the low-level elements, from which the model is constructed, and changes in the outcome of these performance parameters.

We develop the terms of reference for this work starting with some definitions. *Transition* is the change based on some set of actions that moves the object, in this case *technology*. While we cannot draw this thing called *technology,* nor can we draw it, nor sense it, we can associate it with a collection or cluster of thoughts. If we accept that it could be the latter, then it is closely coupled to methods of recognizing and organizing some of its attributes as represented in these thoughts. In this dissertation, we develop a method to measure the patterns of those associations to enable quantification for mathematical manipulation. This leads us to include a key feature, which is a human aspect.

Technologies reflect our human needs. They are mirrors of ourselves. The word *technology* helps us understand this *"process"*. The Greek word τεχνη (or *techne*) describes art and skill in making things. Τεχνη is the work of a sculptor, a stonemason, a composer, or an engineer. The suffix *-ology* means the study or lore of something. Technology is the *knowledge* of making things. Let's put this in a context relative to science and engineering.

The word *science* comes from the word *scientia,* which means "knowledge". We apply the word *science* to ordered and systematic knowledge. A scientist identifies what is known about things and puts that knowledge in some kind of order (Lienhard 2000).

The ordering and systematic collection of information, represented in messages consisting of terms is quantified with a measure in this dissertation.

In its role as the science of making things, technology stands apart from the actual act of glassblowing or machining. It is the ordered knowledge of these things. It is also our means for sharing our knowledge of technique.

Engineering comes from the Latin word *ingenium.* That means "mental power". English is full of words related to *ingenium: ingenuity*, which means "inventiveness" and *engine",* which can refer to any machine of our devising -- any engine of our ingenuity. For about three hundred years, science and τεχνη have joined forces primarily through engineers. Today's engineers are technologists who are well-schooled in science and can make effective use of it when they try to create the engines of our ingenuity.

The three functions of τεχνη, science and invention, work together to make a product. People earn the title *engineer* when the goal of their labors is the actual creative design process -- when they combine the knowledge of τεχνη with science to achieve invention.

A machine normally receives its permanent name only after it has achieved a certain level of maturity -- after it has settled into popular use in the community. Babbage gave a particularly intriguing name to his first programmable computer in the early eighteenth century. He called it an analytical engine. Software packages for checking programs were called parsing engines long before another engine word attached itself to computers: the now common term, *search engine*. We also think of an engine in terms of inputs, some process or transformation and producing some output. This is true of a gas turbine engine, Babbage's analytical engine or a Turing machine. Under stable conditions, an input signal is translated by algorithm into a determinate output. This is how we use the term *engine* in this dissertation. We take an input, transform it into an output using the mental power of the mind, or group of minds in a organization. A physical engine in can be characterized thermodynamically in a mathematical model. This research will develop the properties of the software technology transfer engine model.

2.   Epistemology and Software's Paradox

First, we explore the approaches of science and engineering. As an exercise, establish a mental continuum. On one extreme is philosophy, at the other physics. Philosophy at its extreme is pure logical-mathematical knowledge detached from all experience. It contributes the organizational structures for the experimental, *experiential*, and epistemological search for truth. With the pure philosophical approach, experiential perception assumes frames of reference. At the other end of the continuum is physics at its extreme is a most developed science of experience. It is a perpetual assimilation of experimental fact with logical-mathematical structures. In this approach, we state with sensible experiences and the very refinement of the experience serves as logical-mathematical instruments used as necessary between the subject and the object to be reached. (Piaget 1977, p. 72). For philosophical musings in software engineering, we fall closer to the pure philosophy extreme, but to be practical and useful, we must be able to reach to sensible, physical reality that can be measured.

The problem posed by software engineering is closely related to Planck's paradox. Planck suggested that physical knowledge appears to be based on sensation, and it withdraws increasingly. The reason is that neither philosophy nor software ever proceed from sensation, or even pure perception, but *at the very outset,* it implies a logical-mathematical schematization of perceptions as well as actions exercised on objects. Beginning by such schematization[14], it is natural that these logical-mathematical additions become increasingly important with the development of physical knowledge. Consequently, physical knowledge is constantly withdrawn more and more from perceptions as such.

This is interesting. Software or information cannot be perceived by direct (primary, as defined by Locke) properties, but rather by indirect properties and effects. Let's look at some sensible properties. For example, software has no "mass" or directly sensible weight. This means a basic measure that we might use from Newtonian physics

---

[14] *Schema* is a rule, or category that we use to organize, understand and formulate what we think. (Martin 1991)

is unavailable to us. Software does not appear to have temperature, as a human would sense it. We can't feel hot or cold software or information with our senses. We can not stick a thermometer in and directly measure a temperature. It would appear to not have temperature. Hence, the physical knowledge for software is at the extreme of Planck's paradox *at the very outset*. The observer-scientist developing experiential data is always removed from direct observable property measurement.

This research will suggest that a direct property related to a "volume" can be measured. It suggests that information entropy, and other properties can be calculated. This research will explore property relationships that can be developed using mathematical equations of state.

Software Technology transition, software development and possibly software itself, can be conceptualized as a flow process. Flow processes have gradients of temperature, velocity, and even concentration gradients. A flow system assumes that the intensive properties at a point are the same as if the properties through out the system were uniform and existed at equilibrium at the same temperature, pressure, and composition. The implication is that the equation of state applies locally and instantaneously at any point in the flow system. One may employ a local state concept. In the domain of information, this concept can almost be used as in physics and thermodynamics. The notion of local, however, needs to be extended. In this study local is not defined in physical coordinates, because the medium, a social communication system or network, can communicate influence, or as we said earlier, establish correlation with more geographically remote nodes.

This concept of local state is a universally accepted concept that is independent of the concepts of equilibrium and reversibility. At the very worst, it represents an acceptable approximation.

The models, herein, for software technology transfer, (or in future research, evolutionary development or software), look heuristically at the logical-mathematical schematization of properties (extensive and intensive) for software engineering research equations of state. In this dissertation, we develop an abstract model, a logical-

mathematical schematization, with relationships about information (measured in entropy), a property which cannot be directly measured. Mathematics is performed on the properties. Then we validate the model by taking quantities, which we can measure, e.g., numbers of nodes, the count of terms, and production rates. We transform those measurements into volume, entropy and rate of change of state (the 1st derivative which is like a velocity) publication rate distributions. Then we compare the predicted abstract measures with the observed values transformed to the indirect measure of entropy and frequency.

Piaget develops such propositions, as he explored and traced the psychological origin of notions back through history to their pre-scientific stages. The fundamental notions of physical space, speed, and causality, are in fact borrowed from a common meaning very much prior to their scientific organization. He studied a kind of mental embryology in his development of a theory of knowledge. Piaget eloquently develops a line of reasoning that shows that all the sciences have a common thread. That is, in the process of developing the science or knowledge, there is a fundamental *learning curve*. The learning curve takes on the role of varying the efficiency of a physical system. The learning curve acts as transfer function from state to state of the system.

There are many studies about the proper formulation for learning curves for different problem sets. The majority of the learning curve models indicate that the time to perform a task decreases with the number of times a task has been performed. This is covered extensively in the literature. A review of the relevant historical studies is shown in Chapter II (in 10. Learning Curves, p90). Chapter III develops the learning curve formulations used in this research (Appendix G Learning Curve, p443).

3. Learning Vignette (Meno and Socrates)

Let's start this thread with the discussion of rational analysis. There are many points where one can start the development of the relationship between rational analysis and experiential accumulation of understanding in the reduction of uncertainty. That is *truth* (epistemology) and the *search for truth* (science). The ancient philosophers,

Pythagoras, Protagoras, Socrates, and Plato start the first discourse (the message) that has continued throughout history. Socrates' dialogue with Meno (Plato c428-c348 BC.p163), (Polanyi 1969) addresses an essential question in the search for truth. This is discovery of a distinct type of knowledge: the knowledge of facts of daily life (experiential); and truth, that which has always been and will always be true. With Meno and the Socratic method, we observe immersion, decisions, and a learning process. Socrates did not teach Meno the previously unknown (to Meno) Pythagorean principles for the area of a figure. Rather, Socrates guided Meno via rational thought and decisions through a discovery process. A process implies some type of activity. A process causes a change from one state to another state. Questions were asked and Meno made decisions based on information input a series of symbols, scratching, and utterances. There was a change in the state of Meno's knowledge as he absorbed and combined symbols. There was progress as symbols were put into order and associations were understood. We shall see in Chapter III (B.   Information Theory - Shannon's Entropy, p96), that information entropy is related to the number of decisions that must be made. While the scratching of a geometric figure on the sand was real for the moment, and sensible, it was not the *true form* of a right triangle, but merely a representation or a model of $a^2 + b^2 = c^2$. Examining the dialogue, we see a learning process that included experiential action (observing the figure, and counting). We also witnessed the *progressive accumulation* of understanding as Socrates and Meno interacted (or communicated; Socrates only provided guidance), as Meno did the unpacking of the technology *"message"* from Pythagoras. This process is characterized by accumulation learning, modeled by learning curves in Chapters II and III (10.   Learning Curves, p90, and p443). Part of the effort in reducing the uncertainty (Wehrl 1978, and others) in the message went into unpacking -- or deciphering, and use of a protocol. There is a length of a process (program), which is required to unpack a message (Kolmogorov 1956, Wehrl 1978, Li 1993, Chap 2,3). In this case, the encryption and protocol were the formalisms of mathematics and logic.

The key points this research will develop are all in this ancient vignette -- reduction of uncertainty through discovery, learning, and persistence of a message.

The following chapters of this dissertation will review the literature (Chapter II), develop a model related to evolution of technology (Chapter III), and validate the model based on software engineering technology data (Chapter IV).

4. Communication, Continuity

Communication is a process in which participants create and share information with one another in order to reach a mutual understanding. This definition implies that communication is a process of convergence (or divergence) as two or more individuals exchange information in order to move toward each other (or apart) in the meanings they ascribe to entities (objects, acts, events, etc). (Rogers 1983) Rogers and Kinkaid represent this communication in the general case as a two way process of convergence rather than a one way linear, act in which one individual seeks to transfer a message to another. (Rogers Kinkaid 1981).

This simple concept of human (or machine) communication seems to accurately describe certain communication acts or events involved in technology diffusion.

5. Diffusion

Diffusion is the process by which an innovation is communicated through certain channels over time among the members of a social system. It is a special type of communication, in that the *messages* are concerned with new ideas. (Rogers 1983) For example, when a change agent seeks to persuade a client to adopt an innovation. Examining what occurs in the time step prior to an event and after an event, it is clear the event is only a part of a process of exchange between individuals (or machines). Rogers asserts that it is the newness of the message content of the communication that gives diffusion a special character. The newness implies that some degree of uncertainty is involved.

6. Uncertainty and Confidence

Let's set the context. How do we make choices in the face of uncertainty? We know that a reasonable person having some historical experience with a true coin A, would assign a degree of belief (subjective probability) of about .5 probability for heads. Based on the history with the coin, we would be rather *confident in that belief.* Now imagine a coin B, and we know absolutely nothing about this coin. We don't know whether it has two heads or two tails or if it is a fair coin. Yet, if we had to pick, we would be compelled to assign a single .5 probability, since we lack any information to indicate a greater or lesser belief in heads vs. tails. But, our *confidence* in .5 for coin B would surely be less.

On the one hand, it is not the psychological sensation of confidence that we are interested in. Rather, as an engineer or decision maker, the consequences of the decisions are the driving issue. When we have the option of acquiring information through an *informational* action, we are likely to invest energy (money, effort) before making a decision that results in a terminal action. We would be willing to invest this additional effort in acquiring information about coin B vs. A. So we see that one's *informational* actions, though not one's *terminal* actions, do depend on one's *confidence* in beliefs.

This notion of confidence plays an important role in this discourse's assessment of a software technology.

We are influenced by a number of subjective factors that are always at work. These subjective factors mirror ourselves and often are the emotions of the heart. Beauty and efficiency in art and music, for example, drive human needs as well as functional, quantifiable attributes to reduce the expenditure of labor and effort to achieve a goal. We would be remiss if we did not at least acknowledge the effect these subjective needs have on shaping our technology. The effect of the shaping of technology by these subjective factors which serve the more elemental needs are not that evident by direct observation. There is psychology at work in our methods of acceptance, understanding and ability to assimilate. Often, the reason technology is impossible to predict is that our predictions are inevitably shaped by those factors that are fairly evident. This, therefore, requires

that we address the process of assimilation of knowledge. Using statistics and probability theory, we will stop short of turning this into a study in psychology.

7.   Chance, Aggregation through Mixing

Today we tend to regard knowledge as a process more than a state. This stems partly from the epistemologies of the philosophies of science: The probabilism of the French mathematician Cournot, and his comparative studies of various types of notions set the stage for such an understanding. Critical reviews of historical works, which reveal the oppositions among the various types of scientific thought, clearly promote such a development. Even after the victory of Newton, physics believed for hundreds of years in the absolute character of its principles. So, the arguments developed in this research very much depend on the state and maturity of the knowledge process for software engineering.

Another probabilistic feature of software technology transition is *chance. Chance* is a curious notion which is defined by Cournot as an interference of independent causal series and which generally can be designated under the term *"mixture"*. (Piaget 1977, p. 19) This is an important concept to expose. *Mixture is irreversible and grows with an increasingly weaker probability of return to the initial state.* This starts to address the aggregation typical of composition of terms and integrating domains and technologies.

II.   ASSESSMENT OF PREVIOUS WORK

A.   TECHNOLOGY TRANSFER MODEL FEATURES

Technology transfer (TechTx) or transition is referred to as *diffusion* in the literature.   This section reviews the basics of technology transition models. Various theories and principles felt to be underlying human behavior and learning are presented. The technology transition model basics identified in the literature are then summarized. Seven sections identify research facets or features relevant to technology transfer. These models are shown in Table 1. Table 1 shows the model, a key feature of the model, and indication that the model proposed in this research addresses that feature. Each of these models in Table 1 are summarized in the following sections.

Model In Tech Tx Literature	Model Feature	Proposed Information/Control Theory Model Contribution
Theory of Human Needs Model	Complexity factor framework facts, perceptions, actions	Learning Curve Actions on messages (tasks)
Structure Changes Model–	Internal and External Relationship	Shannon Entropy of Messages Joint entropy Information In, Information Out
Technology Model	Goodness of Technology Alone causes Diffusion	Identifies Minimum number of nodes (senders and receptors) extensions may address vacuum and pressure
Institution Building Model	External Influences affect the human behavior to assimilate a technology	Identifies Entropy as a factor that can influence the acceptance of a technology
Equilibrium vs Conflict Model	Equilibrium is an Instrument for Balance Conflict Is a Instrument to apply Pressure	Convergence in Entropy yields balance Large Differences in Entropy yields Pressure
Communication Model	A Technology is Delivered to Adopters Through a Channel, If Understood It is acted upon.	Quantifies the Information Being Acted on, and Quantifies notion of "Understood " in terms of Entropy and Learning Curve
Problem solving Model	Present Hypothesis Test Hypothesis with Data and Logic	Hypothesizes a Mathematical Model and Explains based on Data Analysis

Table II-1 Technology Transfer Models, Features, and Relation to Proposed Model

1.   The Theory of Human Needs (Leagans 1979)

The theory of human needs (Leagans 1979, p. 15) has a number of components. These are as follows: the facts, the perception of the facts, human attitudes or value judgements about the facts, and human actions related to the facts as they perceive them. Leagans establishes a framework addressing the complexity factors that affect behavior with respect to technology transfer.   The model elements suggested in our current research had to be general enough to permit lower level detailed elaboration that could address these details. This requirement for generality is driven by the need to refine the models to address implementation aspects of technology transfer. The proposed model addresses this through the mechanism of the learning curve and decomposition into organization and sub-organization nodes.

2.   Structure Changes – Internal - External Relationship (Piaget)

While Piaget's[1] work was not focused on technology transfer, his work is fundamental to learning schemes and to an accommodation of these schemes to the environmental situation (Piaget 1963, p. 103). He develops the relationship between the genotype (internal) and phenotype (external) information influences. Yet, neither internal nor external factors can individually explain human development of skills. We can think of this learning in terms of the acquisition of technology. During human knowledge and skill development, it seems to tend toward the establishment of equilibrium of the internal and external factors. (Piaget 1967, p. 113) The proposed *TechTx Entropy Learning Curve* model explored in this dissertation addresses this in several ways. First, the Shannon entropy approach, which takes a vocabulary as input and a vocabulary as output, and from the joint entropy (Bayesian) relationships, yields a grammar. In both the *TechTx Entropy Learning Curve* and *TechTx Entropy Feedback* models, the vocabulary-grammar relationship between internal and external factors is incorporated using Shannon's statistical approach to entropy.   The *TechTx Entropy Feedback* model

---

[1] Piaget, Jean (1876-1980) was a Swiss pschologist.

addresses mixing. It also accommodates structural changes (more explicitly addressing the external factor) due to feedback from external nodes.

3.   Technology Model

The technology model (Leagans 1979) deals with potential. This model suggests that the attractiveness of a new technology alone is sufficiently strong to induce wide diffusion, acceptance and adoption by users. It tends to assume that users would use the new technology and attendant parts of the technology successfully without the persuasions of an organized education system. This model has proven highly inadequate when trying to introduce technology to large masses of users, rather than the elite self-motivated few (Leagans 1979, p. 17). This inadequacy is also consistent with the small percentage of innovators and early adopters identified by Rogers (Rogers 1983 p. 247). However, it does imply that a pressure or a vacuum may have some influence e.g. the growth of the Internet creates a requirement and hence a vacuum, and intelligent agents move in to fill the void. This is analogous to the saying, "necessity is the mother of invention." The current research detailed in this dissertation does not directly address potential or a vacuum. However, the model currently being explored seems to set the stage for future research to be able to see the effects of a vacuum.

4.   Institution-Building Model

The laws of maximum and minimum are often referred to as the institutional factors that explain the forces influencing plant growth. This has been applied to human behavior with the following rationale (Leagans 1979, p. 13): human behavior is the dependent variable. The assumption is that man can influence the economic, biological, and other forms of change to the extent that he controls the forces (nutrients) that influence change and the status quo. In this context, he argues that people see one or more inhibitors (limiting factors) and one or more incentives to innovation simultaneously in any situation. These variables contain and exert varying force or valence on the dependent variable - human behavior - and that when the deficiencies (inhibitors) are weakened or removed, the balance or equilibrium of opposing forces will

be altered. Changes in human behavior are expected to be proportionate to the amount of cumulative influence or valence exerted by the change incentives present. These changes are the net sum of the counteracting influences or change inhibitors operating in the situation.

The model in our study in research uses information theory to quantify the probability via mutual information and , joint and conditional entropy as a method to address the valence of these forces. Further, the current study builds on the notion of need for feedback being proportional to the cumulative influence of the change incentives (information) present. The control model used herein is non-linear. This addresses the comment by Leagans (Leagans 1979, p. 14) that "the input-output function is not always linear." He states that the probability derives from variation in the nature of the influencing factors which vary by situation. For the research herein, we address this by means of an ensemble of very probabilistic primitive communication interactions using both information and control theory.

5.   Equilibrium vs. Conflict Model

In the equilibrium vs. conflict model, equilibrium is regarded as an instrument for achieving balance, while conflict is an instrument for applying pressure.   Some combination of these divergent approaches does in fact operate in most models as a force for motivating people to adopt new patterns of behavior. This is consistent with Piaget and the tendency toward the establishment of an equilibrium of these factors.   In developing the mathematical model of this study, it was interesting to discover that the communication control model used can settle down into equilibrium (oscillating), repelor or attractor stable states.   Oscillation is seen under some conditions of the feedback model. When there is a vacuum, or pressure is applied to a node, learning is more rapid, up to a point. Ultimately each statistical band of nodes reaches capacity. This can be seen in the proposed models.

Prigogine (Prigogine 1980, 1984), who won the Nobel Prize in 1977, says that living (read this as evolving) systems are rarely static, and if they are, they are likely to

atrophy and die from stagnation. Living organisms do not thrive in a state of balanced equilibrium, but usually in fluctuating restlessness. Consumers, organizations, and the technology evolution system itself seem to act as a living organism. The model developed herein addresses these concerns.

6.   Communication Model

The communication model is considered the classical model for diffusion of technology. It is well developed and documented by Rogers (Rogers 1983, 1995). This consists of making a new technology discovery, delivering it to potential adopters through various communication channels, and then being understood and acted upon by the consumer. The communications model is generally seen as a macro model.

Almost every well-researched technology transfer model addresses the communication model. Leagans (Leagans 1979, p. 19) cites Rogers (Rogers 1975), who identified several shortcomings of the model. These include the need to address greater process orientation, greater attention to causality, and recognition that the adoption requires a physical or overt act. This dissertation addresses these shortcomings in the formulation of the mathematical model in section 6. The process aspect is in the message and feedback loops in the control model. Causality and overt act are built into the transforming function $f(x_k)$ in a time step in Chapter III.

7.   Problem Solving Model

This model presents a hypothesis of an explanation of a troubled situation. It tests the hypothesis with data and logic developed putting those specific results into a model. The hypothesis for solving the problem is formulated. Implementing programs and evaluations to assess the consequences tests the proposed solutions. This implementation evaluation/ includes the means and the ends. Boehm and Basili (Boehm 1999, 2000) essentially are espousing that the Department of Defense institute a national effort with Centers for Empirically Based Software Engineering (CeBase) to address transition, using essentially this model.

The current study develops a model at a macro, or strategic, level to predict and plan the technology infrastructure portfolio of a National Technology Transition effort. The current model efforts and elements are reflected in the Department of Defense Software Engineering Science and Technology Summit findings (Boehm 2001).

8.   Classic Diffusion Tech Tx Models (Rogers 1983, 1995)

The Diffusion of Innovation (Rogers 1983, 1995) is one of the most valuable readings on technology transition in general. The approaches of virtually all aspects of technology diffusion are covered. Rogers discusses a communication model that depicts the classic business school "*S*" curve (Rogers 1983, p. 47). This is a cumulative plot of publications covering a given topic over time. Further, he categorizes the four main elements of diffusion of innovations as follows:

- The Innovation
- Communication Channels
- Time
- A Social System

He lays out clear definitions that are commonly accepted in the literature of technology transition and diffusion. Rogers' lexicon can also be seen in the software engineering technology transfer literature. (see Moore 1991, Redwine 1984, Fowler 1994, Fichman 1993, Zelkowitz 1995, and Pfleeger 1999).

Looking at Rogers' work, you can see all of the elements of a communication system. He classifies and distributes the types of adopters (see Figure II-1) as innovators, early adopters, early majority, late majority and laggards. He stresses the uncertainty-reduction aspect of technology. He, as do many, use the terms "innovation" and "technology" as synonyms.
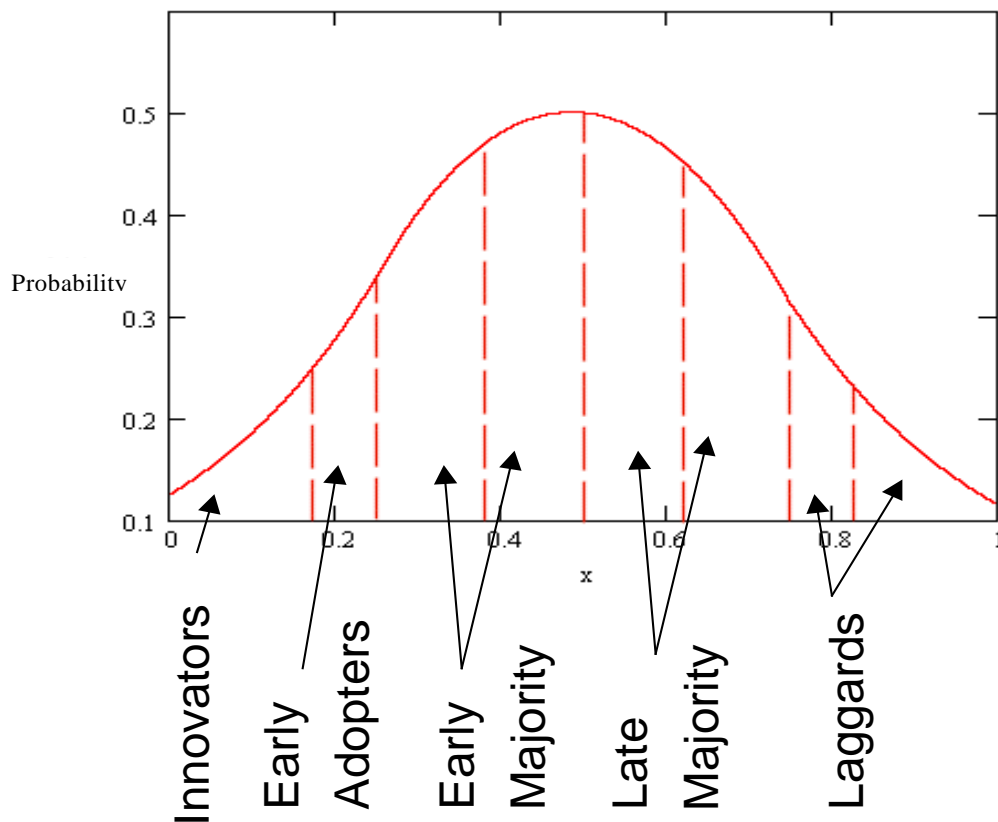
Figure II-1.   Distribution of Adopters.
(Source: Rogers 1983, p. 11).

Rogers identifies *technology* as a design for actions that reduce the uncertainty in the cause and effect relationship involved in achieving a desired outcome. (Rogers 1983, p. 12). The technology developed in the case of this study is itself the technology transfer model. The TechTx Entropy Learning Curve and Feedback models, use a transfer function to represent the reduction in uncertainty and the cause and effect relationship. The proposed model in this research provides a method to analyze options for instrumental actions in order to reduce uncertainty in the arrival of a given set of software technologies.

a.   The Innovation

In the literature, technology generally is seen as having two components, hardware and software. Rogers is speaking of hardware and software in the most general

sense, not limited to computers. 1) *Hardware* consists of the tool that embodies the technology as material or physical objects. 2) *Software* consists of the information base of the tool.

Technological innovation creates one type of uncertainty in the minds of potential adopters (about its expected consequences), as well as representing an opportunity for reduced uncertainty in another sense (that of the information base of the technology itself). The latter is the potential uncertainty reduction representing the possible efficacy of the innovation in solving an adopter's need or perceived problem.

Once information-seeking activities have reduced the uncertainty about the innovation's consequences to a tolerable level, a decision to use the innovation will be made. Figure II-2 shows that the probability of use of various technologies vs. time. As the probabilities of use increases, the risk decreases at a given time. We can see this by analyzing the probability distributions of the consumption of information when observing the output of an organizational unit. We can compare the stochastic dominance of two alternatives accounting for the utility (a function of return and risk) of the alternative.

The models in this research address the innovation-decision process, which is essentially an information seeking, information sending, and information processing process. While this is not directly visible in the *TechTx Basic Entropy* model, the effects of the learning curve are found in the *TechTx Entropy Learning Curve* model. The *TechTx Entropy Feedback* model, working at the organizational and sub-organizational node level, factors in the request for clarification and feedback in order to reduce the uncertainty about the advantages and disadvantages of the innovation.
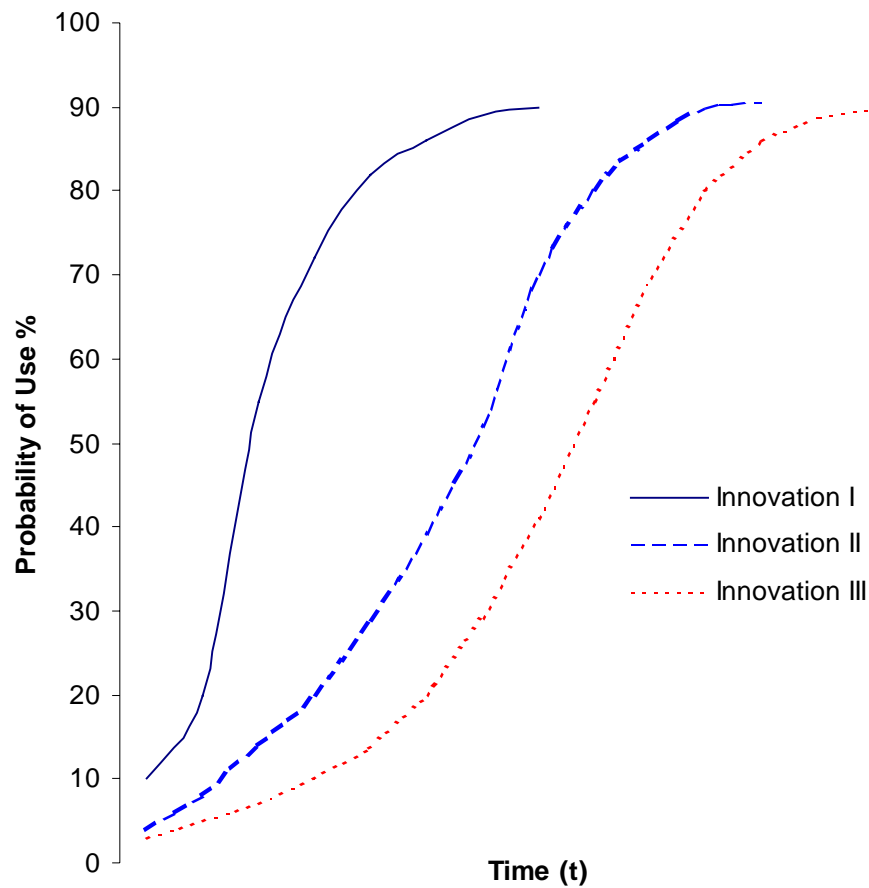
Figure II-2 Diffusion. (Source: Rogers 1983, p. 11)

b.   *Communication*

The primary model in Rogers 1983 is a communication model. While Rogers lays out the communication channel element as component critical to diffusion, he performs and references an enormous amount of empirical data without addressing the model in terms of a communications system. Applying communication and information theory methods to this observation is indeed an area that could benefit the study of software technology transfer. The benefit of an information theory and communication model approach has not been addressed to date. The model developed in this dissertation suggests a quantitative method to address the communication model using Shannon's

entropy, the eigenvalue of the bakers' transformation (an entropy) of the control model and learning curves.

c.   Time

Time is an important element of the diffusion process. Rogers (Rogers 1983 p36) identifies time involved with the: 1) innovation-decision process, 2) innovativeness and 3) rate of adoption of the innovation process.

The innovation-decision process is the mental process that an individual or decision-making unit passes from first knowledge, to forming an attitude about the innovation, to a decision to adopt or reject, to implementation, and finally confirmation or validation of a decision. In Figure II-2, the horizontal distance at a given *y* value of risk/certainty between the upper band and the lower band can be seen as representing this time difference from knowledge to confirmation. Convergence tells us something about the maturity of a technology.

Innovativeness is the degree to which an individual or other unit of adoption is relatively earlier in adopting new ideas. These individual or unit is categorized into one of the five adopter categories.

The rate of adoption is the relative speed with which an innovation is adopted. A steeper curve in Figure II-2 indicates a higher rate of adoption.

Time does not exist independently of events. It is an aspect of every activity. We think in terms of astronomical time, or time differences similar to asking a person on the street for the time and they look at their watch. Rogers and all of the technology transition literature address this type of time. This is time as described in classical physics. We in western scientific tradition take this for granted since the writings of the philosopher Aristotle, in which *time* is closely related to motion and therefore to space. This is a classical interpretation of time in which the present separates the past from the future.

In the basic work *Process and Reality*, Whitehead emphasizes that the simple location in space-time cannot be sufficient and that the embedding of matter in stream of influence is essential (Prigogine 1983). Whitehead emphasizes that no entities,

no states can be defined without activity. No passive matter can lead to a creative universe.

It is only recently that *time* can be expressed in a precise mathematical form. Since we are faced with Planck's Paradox, with the absence of a physical reality, this study moves toward the mathematical notion of time as taken with the use of the bakers' transformation in time steps and presented by Prigogine (Prigogine 1983, 1989, 1997).

The bakers' transformation is essentially the folding and stretching that results in mixing. A summary of the bakers transformation is well described by Prigogine (Prigogine 1989 p200-205). To better understand the function, let's examine two examples normally given to describe the process. Imagine Rome, when we observe the city, we see architecture and buildings from many time periods. They are all interspersed and mixed into the city. These areas and remnants, which are interspersed, are the result of mixing at a number of iterations. The other example, and the one where the bakers' transformation gets its name, is folding and stretching of dough horizontally and vertically. Take a piece of dough, and place a spot of sauce on the dough. Fold the dough. Stretch the dough to be the original area again. Then successively repeat the iteration action. We can let *X* be the function that represents the value corresponding to the application of *n* bakers' transformations.

$$X_{n+1} = F(X_n) \hspace{3cm} (2.1)$$

The various functions X_n are functions of internal time. The internal time is an *operator* like the one used in quantum mechanics. The age of partition X_n is the number *n* of iterations *i* that are to be performed to go from X_o to X_n. Whenever the internal time exists, it is an *operator*, and not a number. The dynamics described by the bakers transformation is conservative, invertable, time reversible, recurrent and chaotic. These properties are the same properties that characterize real-world dynamical systems showing complex behavior Prigogine 1989 p203). Further discussion can be found Appendix A Information, Control Theory and Evolutionary Dynamical Systems Basics, in Prigogine (Prigogine 1983, 1989, 1997), Farmer, York Ott, (Farmer 1983), McCauley,

(McCauley 1993), and Baker (Baker 1990). This is the form of the finite difference equations used in the models.

### d.	Social Structure

The social structure provides the network and media to transmit the messages in the communication-diffusion model. Rogers (Rogers 1983 p. 25) quoted Katz, "It is unthinkable to study diffusion without some knowledge of the social structure in which potential adopters are located as it is to study blood circulation without knowledge of the structure of the veins and arteries." The social system is a set of interrelated units that are engaged in joint problem solving to accomplish a common goal (Rogers 1983 p.24). In other words, the model is a kind of graph.

There is more to it than interrelated units when establishing the network of individuals and organizations. Hargadon (Hargadon 1997) provides an interesting insight via an ethnography on these network mechanisms, for technology brokering and innovation in a development firm that produces one of a kind products. He identifies the mixing mechanisms and the feedback process, building on historical data and experience. The experience is held in informal networks and is communicated in terms that are aggregations and abstractions of terms that were used in prior internal efforts. Typical of the communication were short hand descriptions that would sound like, "We can build this with a *X* like a *Y* from the *Z* project." In this dialog, *Y* is an abstract chunk of a previous project.

Allen (Allen 1977, 1983) and others emphasize the importance of the "messages" from outside organizations. He indicated that as many as 80 percent of the messages come from sources outside the organization. This is interesting since the model proposed draws on external sources of information providing "messages". This is also one of the points of departure from a thermodynamic system consisting of particles. In a thermodynamic system with physical particles, the important feature of stochastic dynamics is the local, short-range character of the interactions. In the physical system, the number of transactions going on per unit time in a system of size *N* must be

proportional to the size. That is each element can only sense its neighbors. In a social system, especially in the technology transfer communications of today, due to the Internet, mass media, telecommunications, fully text and indexed databases, this local character has to be redefined. Local is not geographically local, but rather defined as accessible by a direct contact. Each element can simultaneously sense all of the other elements present. This is addressed in the input to the models developed in Chapter 3.

Another aspect influencing network size in a social system is "who you know" and how efficient, and the endowment of the social network. There is a method to determine effective – efficient network size and diversity, referred to as optimizing *structural holes* of *social capital* (Burt 1992). Essentially social capital is found in relationships – "who you know." It is managed, and it aggregates from people to organizations and can be orchestrated to build an effective social structure and network. The model proposed in this dissertation addresses the node linkages of authors and corporate sources by using the joint entropy of Shannon allocated to performing nodes. While the models herein do not develop these details, the models have been developed to accommodate a structural hole analysis. The approach chosen enables later refinements as detailed node relationships are developed for lower level models, e.g. references cited or actual studies of message traffic to a receiver node.

In competitiveness, or survival, social capital is organized naturally around the human behavior and the principle of least effort. In simple terms, this principle of least effort says that a person solving the immediate problems will be viewed against the background of the person's future problems, as estimated by the person. Moreover, the person will strive to solve the person's problems in such a way as to minimize the total work that must be expended in solving both the person's immediate problems and the person's future problems. That in turn means that the person will strive to minimize the probable average rate of his work-expenditure (over time). And, in so doing he will be minimizing his effort (Zipf 1965, p. 1).

In the area of software engineering, Boehm (Boehm 1989) developed a *Theory W* to help individuals and organizations to negotiate win-win conditions, given constraints and alternatives. *Theory W* is a management theory and approach which says

that making winners of the key stakeholders is a necessary and sufficient condition for an effort's success. (Boehm 1998) First-hand experience by the Army (Saboe 2001a) over the last 10 years with the WinWin process model and tool, indicates that *Theory W* does provide a method for a group of individuals (and by extension this could be seen as representative of organizations) to analyze and act over a larger visible decision space when acquiring a software engineering process technology. This does enable the principle of least effort to be used in a group setting in a more quantitative fashion.

The current research addresses minimum effort through the study of joint entropies in the model. Minimizing the rate of change of entropy, i.e. watching a technology mature, is something that can be observed in the model. On the prescriptive side, actions can be taken to get the technology to stabilize quicker. This is accomplished by investing in refinements, redundancy of the message set, propagation of the messages, increasing the number or quality (performance index) of nodes, and analyzing the effect on the entropy. Hence, the principle of least effort has a place in the model. With the foregoing, we are armed a qualitative discussion of the basics that influence technology transfer. The next section discusses an initial experiment for the software-engineering field to count messages following Rogers' method. This experiment shaped the method that would be developed in this dissertation. Largely, these considerations led this research to a heuristic solution instead of a formal statement of the models.

.

9. Experiment 0 "Count Every Message - Everywhere"

The first experiment, which we refer to as experiment 0, starts to quantify these diffusion concepts for software engineering. The data resulting from the experiment is seen in Figure II-3. Figure II-3 illustrates the message-counting approach of Rogers for the technology. We have the number of messages published in a given year on the *Y*-axis, and time in years on the *X-axis*. Going from the lower to the upper curve follow. The lower curve, is marked with diamonds (◊), Ph.D. Dissertations, Masters Thesis. The curve which is 2nd from the bottom is next, and marked with squares (❏), these are technical reports, proceedings and books. The third curve (2nd from the top) marked with

triangles (Δ) represents articles. The top curve is marked with circles (○), these are also citations from applied science and engineering abstracts.

"Software Engineering" Technology Diffusion
Measured by "Messages" Generated (Saboe 2000)
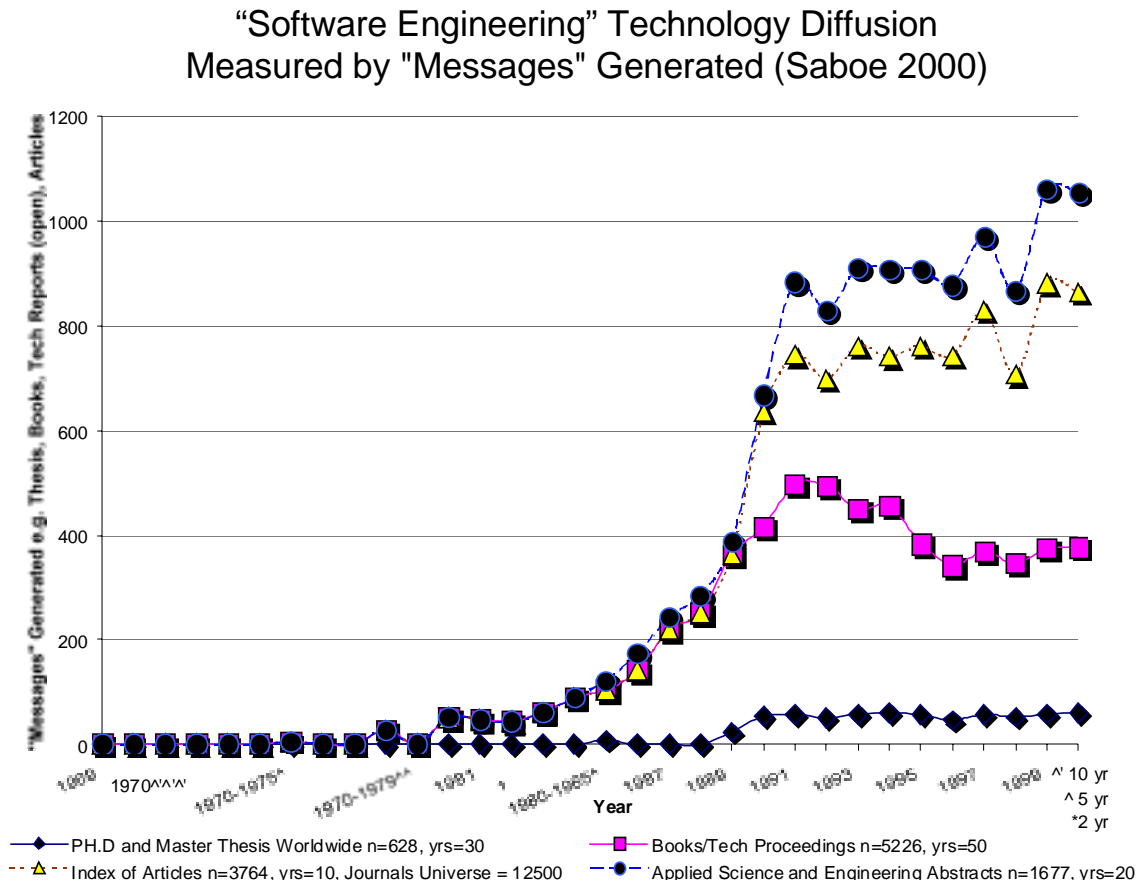
Figure II-3 "Software Engineering" Messages Initial Data.

(Source: Saboe 2000, 2001)

The initial study, called experiment 0, evaluated the technology "Software Engineering"[2] to determine if indeed there was a better way to get a handle on measuring the maturation of technology. During this experiment, the effort looked at all print messages available. Software engineering "messages" were counted starting in 1968.

---

[2] The term software engineering was introduced in 1969, at a NATO conference in Garmisch (Redwine 1984).

The leading indicator messages appear to have grown out of graduate programs that performed research and published messages in the form of Master's theses, and Ph.D. dissertations. Searching Dissertation Abstracts, 628 of these messages were found over a 30-year period. Such messages also appeared in the form of books and technical proceedings. 5226 of these book/technical proceedings messages were found from a source going back 50 years. Messages in the form of articles in abstracted journals had a yield of 3764 messages, over a 10-year period, from a journal universe of 12,500 journal titles. Messages similar to these were searched in another source, the Applied Science and Engineering Abstracts. The result was 1677 messages over a 20-year period. This yielded the data shown in Figure II-3. The data for this chart is found in the appendix. This is a typical message-counting approach. Even when the data is not cumulative, we can see that there are general trends.

We can make a few qualitative observations from the message-count data for software engineering. Looking at the messages published *each year* in Figure II-3, we get a sense of capacity. The research messages from the research institutions seem to be one of the limiting factors. Books and technical proceedings top out as well, also giving an indication of steady state capacity. Articles seem to be still growing. Articles are shorter and therefore more of an overview than the high-end messages in the form of a technical reports, or a thesis or dissertation. The capacity to produce these messages is not as limited. We know, for example, that many papers can come out of one in-depth Ph.D. dissertation. These high-end messages are where one would expect the new ideas to come from. Consulting with researchers, academics, and application developers, there is an intuitive feel that dissertations, thesis, reports and papers, → mostly fuel new research and additional ideas (and create new companies). And that books, some papers → fuel practical applications of research results. Books rarely have new ideas. They have mostly an educational function integrating and restating ideas from the other sources in a form that is accessible to a wider audience. Books also have a filtering function. Books select the most useful new ideas. An informal study done by Potter (Potter 2000) that traced the software engineering topics covered by all editions of Sommerville and Pressman (two popular software engineering texts) observed that as techniques got more

widely used, they were incorporated into the text. Some topics migrated from graduate level course to undergraduate courses, implying a more standard, less complex lexicon.

It is easy to see that the capacity to produce high-end messages has stabilized. The academic research infrastructure is only capable of producing on the order of 100 "new idea" messages per year. Producers of books and technical reports add another 300 messages per year at capacity.   While researchers producing high-end messages containing new information are not the only source of new information, we see they have a capacity limit in the number of messages produced. The capacity limit is expected to change with the nodal learning curve rates. The mind share (similar to market share) fraction of capacity devoted to each subject changes more rapidly. This is visible in the three entropy models. We allocate learning on a per node basis and mind share is reflected in the number of nodes. Rogers attributed the rapid rates of adoption in a technology to more nodes. In order to build a nationally competitive infrastructure, these are the types of leverage points to which research managers and government policy makers need to have access.

While this is interesting, the message-counting approach is limited in its analytical value.   It is a very labor-intensive effort to count every message with minimum quantitative yield that would enable better-informed decisions for proactive actions.

The idea to find a representative sample of messages for the technology under examination pointed to professional societies. While their databases would not cover every message, they would yield a rich enough source to potentially bear meaningful, statistically representative fruit.

10.   Crossing the Chasm (Moore 1991)

Moore (Moore 1991) identified a chasm between the early adopters and early majority. Fissures were identified between the other adopter segments of the communiy. At least two factors contribute. First, the communication channel between the segments of the community may be non-existent or spotty. Second, if the communicaiton channel

existed and was established, there is an impedence mismatch between advocates and receptors in different communitiy segments. One could speculate that if a model was developed that included a notion of momentum, then conditons could be arranged so that enough momentum, with momentum developed from the entropy data, could enable "jumping" across the chasm and fissures due to potential and pressures. This notion of momentum is defferred in this dissertation to areas of future research, but the models developed may have a momentum property.
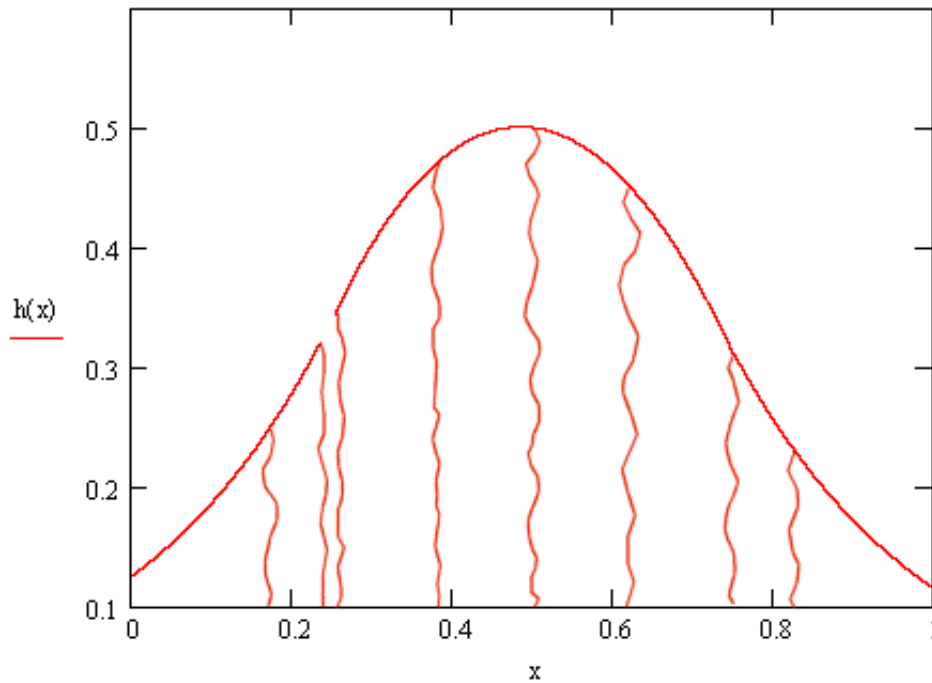
Figure II-4 Crossing the Chasm (Moore 1991)

11.   States of Software Technology

Redwine et. al. (Redwine 1984) studied 14 different cases in considerable detail. They identified 5 major phases, and 2 sub phases, in popularization (4a and 4b), that a technology passes through as it matures. Figure II-5 shows the states. While the analysis is extremely good for the cases studied, there is a bit of imprecision in states 4a and 4b, e.g. popularization throughout 40% and 70% of the community respectively. It is

extremely difficult to determine, based on their methods, how to identify what the quantity for the total community is.

For example, citrus fruit was known to cure scurvy 200 years before the British merchant Navy adopted the practice. The Royal Navy, took 400 years to adopt the practice. One would think it was the same community. Yet, the Royal Navy could impress sailors and really wasn't concerned about attrition, so their impetus to adopt was quite late. At the same time, the merchant navy had a different set of realities. By most standards, we would think of this as one community. In the software community, there is also a spectrum. The realities of resources constrained systems in the embedded world have kept that community from adopting techniques like CORBA in the general purpose processing world of management information systems.

They also make a flat statement that a technology matures in about 7 +/- years. It turns out that this is a very difficult statement to support. On the other hand, they identified several points where we can observe output.

We can observe a report or paper (a message) that identifies when there is a problem that exists (phase 0). The observable facts in concept formulation (phase 1) are general publication (messages) of solutions to parts of the problem. Innovators, in Rogers' terms, would generally be found in phases 0 and 1. Clear definition of a solution via a seminal paper (a message), or demonstration system is the marker for the phase of development and extension (phase 2). While a demonstration system is generally documented in a paper or report, which we can count in the proposed method, a demonstrator is still a message. Internal enhancement and exploration illustrating usable capabilities which are available is a message (phase 3). In phases 2 and 3, you would expect to see the early adopters. When the technology is used outside the initial development group (phase 4), we see more observable messages. This is also where it moves to the broader consumer community. It is at phase 4 that the early majority and late majority are generally observed using the technology.

Each of these observations can be viewed as a message. More particularly, these messages are reported in the literature, which is professionally indexed and abstracted.

During the validation of this research, data has been gathered on five of the fourteen technologies in addition to more current technologies. While there may be a method to map these state transition points that are clearly observable to entropy curve characteristics[3], i.e. 1st and 2nd derivatives, as well as inflection points, stochastic dominance, etc., this has yet to be done.

States of Software Technology Transition
(Redwine 1984)

b -- throughout 70% of the community

a -- throughout 40% of the community

Popularization
• appearance of production quality,supported versions
• commercialization and marketing of the technology
• propagation of the technology through a receptive community of users

Substantial Evidence of Value and Applicability

4 Enhancement and Exploration (external)
• Same activities as for *Enhancement and Exploration (internal)* but -- they are carried out by a broader group, including people that have not been involved in the technology maturation up to this point

Shift to Usage Outside the Development Group

3 Enhancement and Exploration (internal)
• major extensions of the general approach to alternative problem domains
• use of technology to solve real problems
• stabilization and porting of the technology
• development of training materials
• derivations of results indicating value *Usable Capabilities Come Available*

2 Development and Extension
• trial, preliminary use of the technology
• clarification of the underlying ideas *Clear Definition of a Solution Approach via a*
• extension of the general approach to a broader solution *Seminal Paper of a Demonstration System*

1 Concept Formulation
• Informal circulation of ideas
• convergence on a compatible set of ideas *Appearance of a key Idea underlying the technology*
• general publication of solutions to parts of the problem *or a clear articulation of the problem*

0 Basic Research
• Investigation of ideas and concepts that prove fundamental to the technology
• general recognition that a problem exists and discussion of its scope and nature

Figure II-5. States of Software Technology Transition. (Source: Saboe 2001, Redwine 1984)

12. Software Technology Transition Framework, Advocate/Receptor

The Software Engineering Institute has been the single most prolific source on the subject of software engineering technology transfer. This is readily understood since this Federally Funded Research and Development Center was established with a primary mission to establish transfer of software engineering technology to the Department of Defense. Fowler (Fowler 1994) developed a framework for technology transfer

---

[3] Any undergraduate calculus book tells us that setting the 1st derivative equal to zero, determines whether a local maximum and minimum exist and the location. Setting the 2nd derivative equal to zero identifies the inflection points. Those points and the 1st and 2nd derivatives show the characteristic of the curve. This shows how the slope changes, as well as how the curve bends upward or downward.

identifying advocates and receptors (change agents) mediating between producers and consumers (see Figure II-6). In this work, three life cycles of technology transition are presented: research and development, new product development, and implementation. Emphasis on the need for common terms between receptors, consumers, and researchers is identified as an important aspect of the SEI studies. This dissertation's model accounts for this finding by examining the conditional probability, e.g. the input terms influencing the output terms (See 4. Conditional Entropy p107). A clear signal, with minimum noise and need for requests for feedback, between a sender and receiver improves technology transfer.

Software Technology Transition Framework
Producer Consumer Model with Advocates and Receptors
(Fowler 1994)
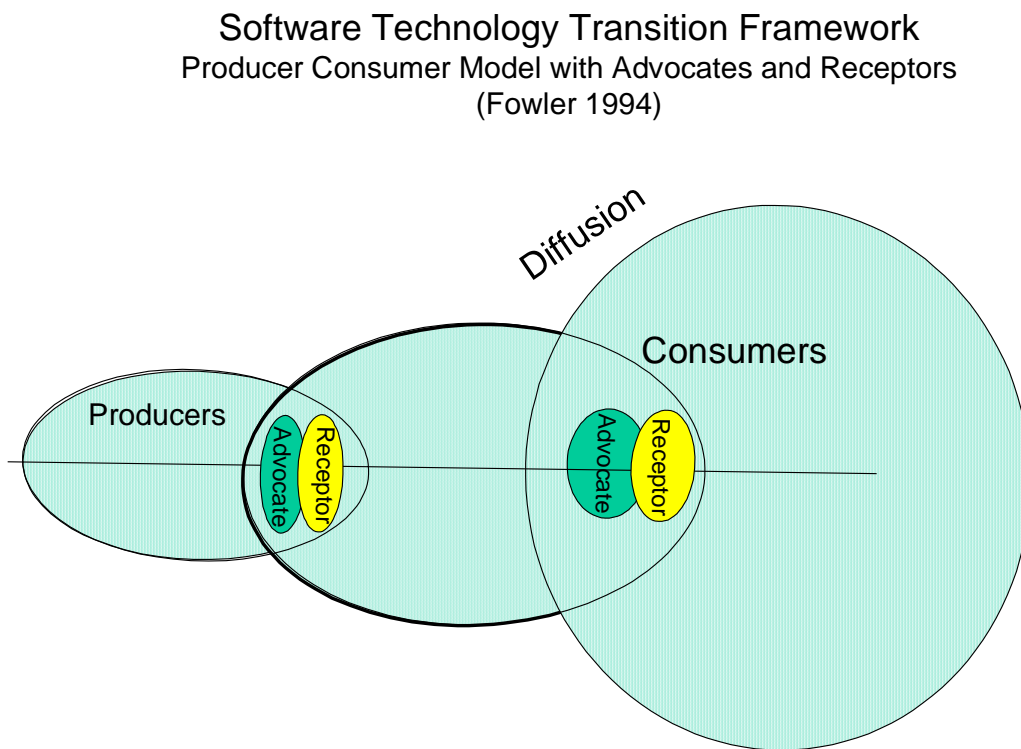
Figure II-6. Software Technology Transition Framework. (Source: Fowler 1991)

This research does not address the lower level implementation details of that framework; rather it builds an analytical framework useful to determine probability of success and quantity and redundancy of messages that need to be sent as a clear signal.

Significant additional work (Forrester 2000, Fowler 1992, Fowler 1992a, Fowler 1990) has been developed at the SEI. This work primarily focuses on the lower level implementation details of the framework, e.g. methods on how to plan and effectively communicate technology to an organization.

Saboe (Saboe 2001) has related the framework of Forrester to the early phases and state transition points of Redwine and Rogers (See Figure II-7). Producers are generally in the early phases (0-2) of Redwine's model. Early adopters are in the phases from 1 to 3 of Redwine's model. The early majority, and the consumer picks up from phase 3 through the late majority and other consumers of the technology.

States and Producer Consumer Software Tech Tx Model (Saboe 2000)

Consumers

Diffusion

Receptor
Advocate

tantial Evidence of Value and Applicability

Outside the Development Group

bilities Come Available

Clear ... olution Approach via a
Seminal Paper of a Demonstration System
Appearance of a key Idea underlying the technology
or a clear articulation of the problem

21 June 2001   M Saboe   8
Monterey Workshop 20001
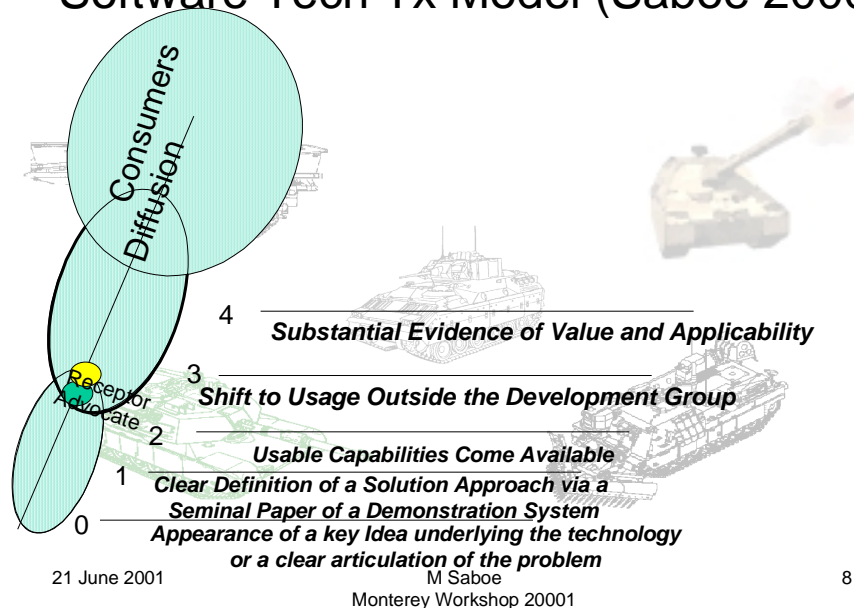
Figure II-7.   Mapping of the SEI Transition Framework and Redwine's Stages.

(Source: Saboe 2001)

13. Thermodynamics Example in Technology Transition States

Let's review some of the history of where we are with regard to a technology that is very relevant to this research--Thermodynamics. As we know, the gestation period for

this "technology" took well over 100 years. Figure II-8 uses thermodynamics as an example technology mapped to the states as identified by Redwine.

Thermodynamics can be defined as the science of energy. (Çengel 1989, p.2). Energy is viewed as the capacity to do work or as the ability to cause changes. One of the most fundamental laws of nature is the principle of *conservation of energy*. This states that the total amount of energy is constant. Thermodynamics deals with conversion of energy from one form to another. It deals with properties of elements under study and the changes in those properties as the result of energy transformations and interactions. During an interaction, the energy in a system can change from one state to another. Thermodynamics defines a control volume, boundaries etc., that represent the system under study. It turns out that the principles of thermodynamics can be applied to any conserved property, e.g. energy, momentum, mass. This is now covered in many undergraduate texts on thermodynamics and physics (Fraundorf 2000). It is useful to apply these principles to information as well. Either the information is conserved and useful, or noise (entropy). This section develops similar properties for software technology transfer. We call this Technology Transition Dynamics (TechTx Dynamics). The principles are constructed in such a manner to support extension to software development and software itself.

It is useful to spot key points for the development of thermodynamics. It was not until 1700, when Newcomen and Savery were developing the steam engine, that the need arose for studying the problem. The first clear articulation of a problem was in 1700. (*State0, "Clear Articulation of Problem"*) The first seminal paper occurred in 1849. This is when Lord Kelvin published the term "thermodynamics". *(State1, "Seminal Paper")* Rankine published the first textbook ten years later, in 1859. *(State2, "Usable Capabilities Come Available")* Practical development *(State4, "Substantial Evidence of Value and Applicability")* and is evidenced in the early 1900s. Gibbs, in 1902 with his *"Elementary Principles of Statistical Mechanics",* Fowler and Tolman in 1936 and 1938 and their publications, *"Statistical Mechanics"* and *"Principles of Statistical Mechanics"* respectively. It can easily be argued that by 1953-54, thermodynamics had reached *State4b, "Popularization Throughout 70% of the Community".* Popular texts by Shapiro,

"The Dynamics and Thermodynamics of Compressible Fluid Flow" and Lee, *"Theory and Design of Steam and Gas Turbine Engines"* saw widespread use for decades.
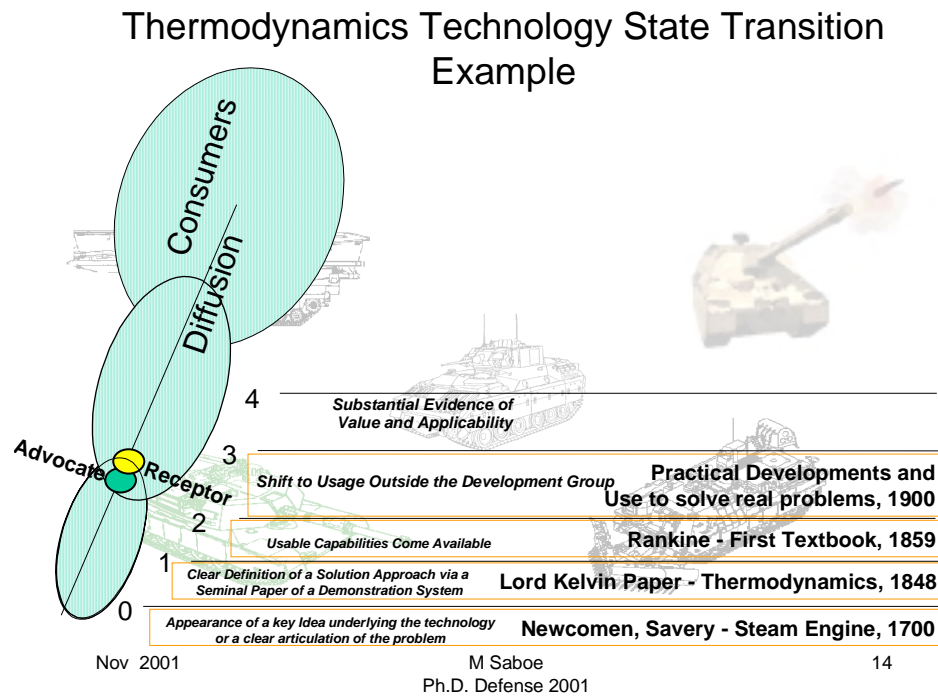
Figure II-8 Thermodynamics Technology Transition State Example

For the purposes of the domain of knowledge for software engineering (technology transition dynamics), the key *State1, "Seminal Paper"* state transition point occurred with Shannon in 1948. Claude Shannon is considered the founder of information theory. He is regarded by some as a modern equivalent to Newton. Shannon picked up the thermodynamic notion of entropy and applied it initially to communications theory[4]. This theory is the underpinning of modern information theory. This can be seen in the top block of Figure II-9.

---

[4] We saw a hint of the future in the 1959 statement by Lyapunov after noting Shannon's work.

"Описанные работы представляют собой первые шаги в области математических эадач кибернетики. Они объединены некоей общей направленностью замыслов, которую можно характеризовать как начало разработки общей метрической теории алгоритмов или теории алгоритмов с оценкамию Однако построение такой теории является ещё делом будущего".

Software Technology Transition - State Example

Consumers

Diffusion

Advocate **Receptor**

4

3

2

1

0

Saboe, Software Engineering 2001
Prigogine Dynamical Systems, Information, Evolution 1980 — 2
Kolmogorov Complexity 1964
Jaynes, Information Theory, Statistical Mechanics, 1957 — 1
Shannon, Communication Theory 1948
Fowler, Tolman, Statistical Mechanics 1936,38 — 0
Clausius, Entropy 1850, Boltzman 1860s, Gibbs 1902

Substantial Evidence of Value and Applicability — *Important Parallel developments, Bernoulli 1713, Bayes 1763*

Shift to Usage Outside the Development Group — **Practical Developments and use to solve real problems, 1900**

Usable Capabilities Come Available — **Rankine - First Textbook, 1859**

Clear Definition of a Solution Approach via a Seminal Paper of a Demonstration System — **Lord Kelvin Paper - Thermodynamics, 1848**

Appearance of a key Idea underlying the technology or a clear articulation of the problem — **Newcomen, Savery - Steam Engine, 1700**

Figure II-9 Software Technology Transfer – State Transition Example

There are several tracks that finally converge to get us to the point of this research. Thermodynamics converges with information theory, and in this research, we tie together thermodynamics, information theory, control theory, dynamical systems, and learning curves with software engineering technology transfer. Later, we suggest that the nodes, arcs, and entropy measure are relevant to software development and software itself.

One of the drawbacks of this view is that we know that everything listed in the upper area of Figure II-9, is primarily the result of work by investigators outside of the thermodynamics community. In the upper block, we have the convergence and mixing of

---

"The efforts described here represent the first steps in the area mathematical problems in cybernetics. They are linked together by some common idea, which can be characterized as the starting point of the development of the common metric theory of algorithms, or the theory of algorithms with estimates. However, the development of this theory is still to be accomplished in the future".   (Halstead 1977, p4, Translation Bankowski, 2001). We will run into Lyapunov again.

several threads of technologies from different domains. We see the probability work by Bernoulli and Bayes in statistics, which has its own set of state transitions. We also see thermodynamics and information theory as finally dynamical systems inspired by biology and evolution of life itself.

Yet there is definitely a foundation laid by the thermodynamics work. On the other hand, if we start with Shannon, we can see a parallel set of states (shown on the right of the figure as local state transitions) using the thermodynamics foundation as an input.

With his publication of *"A Mathematical Theory of Communication"* the discipline was provided a crystallizing and focusing seminal paper. The precursors to this at *State0,* with the *"Clear Articulation of the Problem and Appearance of Key Ideas Underlying the Technology"* in communications, information and mathematical theory was the work by Bernoulli (1713), Bayes (1763), Gibbs (1902, 1928), Szilard (1929)[5], von Neuman (1944), Kolmorogrov (1956), Jaynes (1957, 1957a), Kulch (1972), Uspensky (1992), and Li (1993). We consider the use of these developments in this research is, *prima-facia,* evidence that those technologies have had substantial evidence of value and they are being applied by an outside group – software engineering.

14.   Extension to Address Standardization Effects (Fichman 1993)

Fichman and Kemerer (Fichman 1993) focused on organizational and community-wide technology adoption. They develop a two dimensional framework based on theories relating to organization and communities. They particularly bring the economics of standardization to the literature for the first time in the software engineering process technology literature. This work points out four economic factors affecting technology adoption: prior technology drag, irreversibility of investments, sponsorship, and expectations. These are summarized as follows:

---

[5] It is of interest to note that the Szilard engine was described in 1929 *z.Phys. 53 (1929) p 840-856* according to Zurek. (Zurek 1989). In this paper titled "On the decrease in entropy in a thermodynamic system by the intervention of intelligent beings" he discovered the relationship between information and entropy.

a.   *Prior Technology Drag*

A prior technology provides significant benefits because there is a large and mature installed base. The research model of this research enables the quantification and the detection of "pushback" by measures of the entropy. e.g. the terms of the technology show up more and more in the community lexicon. The models proposed suggest that the more familiar the terms, the less likely the technology will be resisted, or pushed back (Zipf's law of minimum effort[6]) and the fewer requests for clarification will be required. This research is explicitly going to show the relationship of entropy, state transitions and frequency of performing a task (producing a message) (See 3.   Two Dimensional Finite Difference Representation of S_{H_k}, p141.   We know from other learning curve studies, the more times a task has been performed, the less time required to perform the task. This learning represents an increase in performance efficiency. This too is closely related to the law of minimum effort. This research suggests that in the *TechTx Basic Entropy* model, the measure of entropy, as input, gives a synthetic metric for the technology drag.

b.   *Irreversibility of Investments*

Adoption of the technology requires irreversible investments in areas such as products, training and accumulated project experience.   In the section of the Introduction, 2.   Context and Overview, p9 the flow of correlations yields irreversibility. For example, once the money is spent on a technology, it is gone. It can not be spent again.   Another example is closer to the thermodynamic aspect of irreversibility.   Once the community or a node in the community is exposed to a technology, you can not unexpose them. The future is influenced by that exposure to a

---

[6] Zipf's law of minimum effort is really a social structure representation of thermodynamic and Newtonian principles. "Which says to pass from the initial position [or state] occupied at instant t_0, to the final position occupied at t_1, the system must describe a path that in the interval of time between the instant $t0$ and t_1, the mean value of the action – the *difference* between the two energies T [kinetic a function of mass and velocity] and U [potential energy depending only on the coordinates or structure] must remain as small as possible." (Poincare 1903 p63).   Similarly Bayes, using geometric methods, makes a similar argument but in terms of probability, expectations, and variance. (Bayes 1763) In the later case, Bayes, and former, Zipf, there is no reference to the materiel under examination. This reinforces that the principle is not limited to physics and can apply to information correlations as well.

product, training and prior experience. This dissertation prior experience, training and exposure through the entropy aspect of the model. In the control theory part of the model, the requests for feedback become less if the input messages represent well-understood messages by the resources and assets in the node.

c.   *Sponsorship*

Fichman suggests that strong sponsorship seems be beneficial in moving a technology to standardization when a single entity (person, organization, consortium) exists to define the technology, set standards, subsidize early adopters, and otherwise promote adoption of the new technology. The models in this dissertation reflect that conjecture in two ways, one explicit and the other implicit. Explicitly, if the terms in use have been widely accepted as standard, this reduces the noise in the producer- (advocate–receptor)-consumer lexicon, increasing the mutual information used. This reduces the rate of change of the entropy. Also, large quantities with a limited amount of new terms introduced published each year, would reflect sponsorship. Even if there were not a single entity with resources focused to promote the technology, the models would suggest that the technology is approaching stability, and converging. While the model does not address resources explicitly, the result of resource expenditure is seen in messages. A mass of messages with the same vocabulary reduces entropy, moving the vocabulary toward stability. Additional new messages with new terms in the vocabulary at a rate greater than the usage of the existing vocabulary retards the movement toward stability and convergence. Let's look at a sponsor that is providing resources for a given technology. Researchers knowing that there is a customer will direct their efforts to producing messages in the desired technology's lexicon. They are reacting to a potential. It takes time and effort to produce the messages. The more heads, resourced in a band, which address the technology issue, thanks to sponsorship, will yield greater message output. The change in entropy, as the result of new messages in the result of effort, implies resource consumption to produce the messages. The stability and convergence (i.e. decrease in the rate of change of entropy) suggest the lexicon is becoming standardized. This may be defacto. The vocabulary, communication network approach

and the change agent (sender – receiver) aspect of the model address this factor which was seen as desirable and identified by Fichman and Kemerer (Fichman 1993). More interesting is that by exercising the model by varying the number of performers in a band or mix of a portfolio of bands will affect the modeled output. This analysis would suggest the prescriptive remedy a program or research manager should take to reduce risk or accelerate the arrival of a technology.

d.   Expectations

Technology benefits from an extended period of widespread expectations that it will be pervasively adopted in the future. This research sets up the ability to further analyze the notion of expectations and deals the expected value of terms in a technology (1. Entropy Review, p98). The inference is that the more likely that a set of terms is expected to be found related to a technology, the less uncertainty there is relative to those sets of terms and that subject technology. This reduces risk, and increases the probability of use – if the technology is useful for the problem at hand. However, this is the topic for further research as identified in the final sections. Work addressing mathematical concepts of momentum and potential can be developed based on the elements of the initial model.

The work by Fichman and Kemerer also identifies attributes of innovations. Although Rogers addressed and identified five generic attributes of innovation (1) relative advantage, (2) compatibility, (3) complexity, (4) trialability, and (5) observability, his work is based mostly on study of individuals. Van de Ven (Van de Ven 1991) argues that these same innovation attributes play an important role in adoptions by organizations. The Rogers' attributes have been generally adopted by the community. This appears to be due to familiarity (correlations of terms) with the attributes in the diffusion of innovations community. Others (Moore 1987), (Kwon 1987) use these as well. Alternate taxonomies show up in Leonard—Barton (Leonard-Barton 1988).   They identify transferability, organizational complexity, and divisibility. Pennings (Pennings 1987) identifies concreteness, divisibility and cost. Eveland and Toratzky (Eveland 1990) identify trialability, lumpiness, adaptability, degree of

packaging, and the "hardness" of the underlying science. Zelkowitz (Zelkowitz 1998) relates different styles to Rogers' attributes and characteristics of the adopter type. In most cases, all of these can be mapped back to Rogers' original attributes.

This research was constructed to address Rogers' compatibility, complexity, trialability and observability in terms of the entropy metric. The entropy, specifically conditional entropy, addresses complexity of a technology and expectation of adoption. The trialability is inferred from the production index of the number of observable messages produced (i.e. messages produced per time step by a node). This research explicitly models the notion of Rogers' complexity as the entropy of the set of sets of terms that a node takes as input. This research also explicitly relates the production index and the input entropy intuitively this is related to trialability. The more the portfolio of nodes can produce per time step the more trials were performed (based on research task produced per researcher capita). This relative advantage is addressed only indirectly, but the mechanism is there to compare two or more competing technology entropy metric curves and to determine the rate of change, crossover, and probability of arrival of a technology's maturity. Observability at the system level can also be seen in the selection of technologies studied. The data from the technologies studied permit future spotting in of Redwine's observable (first four) state transition points. This represents five of the fourteen technologies Redwine studied. It is premature to say that we can make any predictions by spotting observable points alone. However, future research could spot the observable events and attempt to correlate probability of success with the entropy metric.

15. Diffusion/Infusion Issues (Zelkowitz 1995)

Zelkowitz (Zelkowitz 1995, 1998) has extensive experience with infusing technology into organizations. Infusion is differentiated from diffusion as it relates to internal adoption by a particular target organization, while diffusion generally refers to movement of the technology to the broader user community in a macro sense. His study within NASA builds on the "experience factory" work with NASA's Software

Engineering Lab and the experimental approaches of Basili (Basili 1994, 1994a). He studied the differences in the industry-wide phenomenon of a technology specifically focusing on the infusion process, which actually make the changes in the current state of technology. The TechTx Entropy Feedback model (p148) provides a mechanism to address the infusion process in the transfer function. The function takes as input new messages and interactions resulting from feedback and produces output. Successfully retransmitted messages from a change agent (receptor) to a consumer represent infusion in a particular organization. The feedback model is abstract, but is constructed in a way to permit lower level, implementation details to be added which address infusion. The fraction of messages that need clarification, (β) (introduced in the *TechTx Basic Entropy Feedback* model) in the feedback model, represents a kind of efficiency of the infusion process. The percentage (1-β) of the world messages related to the material is well understood in highly encrypted messages, and without a lot of noise, the technology is passed directly to the consumer. At the macro diffusion level, looking at the entropy rate of change for the ensemble of nodes, we see the associated clarification ($\beta's$) which give us the average rate for the request for feedback (lack of understanding) of a technology. This in turn can be fed to infusion, where the technology program manager and adopter organization can further study the details of the infusion process. Individual β values for an organization and a given technology can be measured, if it is so desired.

16. Technology Transfer and the Learning Curve (Nishiyama 2000), (Hanakawa 1998)

During infusion, there is evidence that the learning curve is in play. The skill level and the improvement in productivity due to the technology, productivity loss during transfer, and the combined effects, net gain (Nishiyama 2000). The learning curve impacts on assimilating a new technology into a project were seen by the number of tasks performed over a study of several projects (Hanakawa 1998). This study in software development and others suggest the learning curve of Newell and Rosenbloom (Newell 1981) for power law *chunking* is appropriate for the various types of learning that need to be handled. This research looks at the learning curve as a local, process efficiency

function to refine the basic control model with the power law learning curve. This can be extended to a power law that uses the chunking model equations. While this is not important for the development of the basic model in this research, it provides the linkage to all manner of studies of organizational learning and ultimately, the breakeven and return on investment curves (Nishiyama 2000). This can be developed to make resource decisions, both for the infrastructure and for a specific research program or organization.

There is a broad base of literature on learning curves. During the study for this research, a large number of papers were reviewed. (Anderson 1981, Guiliksen 1934, Knecht 1974, Langley 1981, Lewis 1981, Mazur 1978, Newell 1981, Nembhard 2000, Miller 1956, Vigil 1994, Yelle 1979) and many more.

These papers developed the basic relationships from learning curves, through relevance to software engineering. Anderson (Anderson 1981) is from Carnegie Mellon University, and the book he compiled under NSF and DARPA funding has a strong bent to showing the relevance to software development. (Langley 1981), (Lewis 1981), (Newell 1981). Linkage to distributions of terms and statistics of language and Zipf's law for the principle of least effort, are connected through (Mandlebrot 1953), (Simon 1955), (Snoddy 1926), and (Zipf 1949, 1965).

17.   Mapping of Motives of Actors (Pfleeger 1999)

While the work by Pfleeger (Pfleeger 1999) never explicitly defines technology transfer, it provides the most comprehensive literature summary of the essential software technology literature. While not addressing all of the transfer field literature, or even all of the software technology studied in this area, the paper is an excellent review, a great overview and starting point. There are several key contributions beyond the survey of the field. She describes the process and roles involved in order to move technology in a transition from idea (technology creation) to adoption (technology diffusion). The generation of evidence, packaging, support and attention to the audience are identified as essential elements in the process of transfer. In this research, these characteristics are primarily addressed in the clarification (β) in the control model. The clarification (β)

values are driven by the commonality of terms to the audience measures in terms of the frequencies and entropy metric.

Pfleeger also maps the motivations of the adopters to the category of adopter (innovators, early adopters, etc. per Rogers 1983) (Table II-2). Also identified are the effects of rules imposed on an organization, a standards committee or a customer. These rules can encourage the success of a technology (this push or pull) when other models fail. For instance, she cites the effect of the Department of Defense's endorsements of products, recommendations for process improvement, or mandatory rules about tools as a positive influence to encourage "laggards" to take risks and try new technologies. The successful technology requires not only a new idea, she claims, but also a receptive audience with a particular adoption style. The various models (people mover, communications, on the shelf, vendor and rule as introduced by Pfleeger) are mapped to the level of risk the adopter community is willing to take.

Adopter Category	Level of Risk	Adopter Model
Innovators	Very High	People-mover model
Early adopter	High	Communication model
Early Majority	Moderate	On-the-shelf model
Late Majority	Low	Vendor model
Laggards	Very Low	Rule model

Table II-2 Relationships among Adopters, Risk and likely Transfer Model.
(Source: Pfleeger 1999)

So to reduce the impedance mismatch between researcher and the method of moving the technology, "message" has to be matched with the audience. While Pfleeger cites Zelkowitz and other studies that look at the actual implementation details of the transfer process, it is useful to note the factors that affect clarification requests (β) in this research. Another way to view the stream of messages is to suggest all that does not move to the consumer is in the feedback-entropy streams. Pfleeger, Zelkowitz, the SEI and others generally are looking at the implementation details of technology transfer. All of the research to date generally looks at technology transfer from this perspective. This

research addresses a macro process, useful to the research manager and program managers, to assess the risk of the technology maturing at a given time. Implementation in a specific program of a technology should try to minimize the clarification requests (β). Using messages that are matched for the audience minimizes the mismatch. The message is packaging of the evidence. Pfleeger (Pfleeger 1999) and Schum (Schum 1994) describe evidence.

Types of Evidence	Characteristics
Tangible	Objects
	Documents
	Images
	Measurements
	Charts
	Relationships
Testimonial (unequivocal)	Direct Observations
	Second-hand
	Opinion
Testimonial (equivocal)	Complete equivocation
	Probabilistic argument
Missing tangibles or testimony	Contradictory data
	Partial data
Authoritative records or facts	Legal documents
	Census data

Table II-4.   Messages in Forms of Evidence.
(Source: After Schum 1994, Pfleeger 1999)

Schum presents the categories of evidence seen in Table II-4. The specific observational sense, objectivity and veracity of the message enable decisions to adopt or not adopt. In terms of this dissertation, if message is clear, unambiguous, and well understood, the advocate can pass on the message to the receptor with little to no requests

for feedback. Schum and Pfleeger argue for this packaging of the message. This research supports those observations with the Shannon entropy component where noise and non-signal are minimized, e.g. the vocabulary converges between advocate and receptor.

In this research, we consider the message as representative of the evidence. The risk is related to how often the terms in the message are expected to be used together by the advocate (publisher of the message) and receptor (consumer of the message). For example, we regularly read papers that give messages representing evidence that a subject technology combined with some other characteristic associated with the technology which was used, examined, etc. The more frequently we these pairs of terms characterizing the use, examination of the technology, the more likely we would expect to see this combination in the future.

Let's consider a message representing evidence as a set of terms, for example the set of terms {}, {A}, {B}, {C}, might be a message about technology {A} with technologies {B} and {C}. The {} represents a null set in this alphabet for completeness. We will see papers, which are a way of transmitting a message, where there are combinations of these terms used. This alphabet can become a type of artificial language, with various combinations of the terms. Potential single combinations are shown in Figure II-10. Sub totals for q-level = 2 is seen as equal to 6, and q-level =3 is equal to 1. For q-level = 1, all of the combinations are the same, while the count is equal to three, there is really only one possibility, null. We will find that we can not count three instances of null, nor can we count one instance of nothing.

Terms	{}	A	B	C	
q=1	{}	o	1	1	1	3

		{}	A	B	C	
q=2	A	o	o	1	1	2
q=2	B	o	o	o	1	1
q=2	C	o	o	o	o	0
						6

		{}	A	B	C	
q=3	AA	o	o	o	o	
	AB	o	o	o	1	1
	BC	o	o	o	o	0
						1
						7

Figure II-10 Potential Single Combinations.

In the first section labeled q-1, we see pairings of the null set and the single terms. This yields a set of sets of singles which are possible to be found. In the second group, q=2, we see the pairings of the single set from q=1 with the primitive set terms. The "o" indicates we are not counting this combination because it is not unique and has been counted already. For example, {AA} tells us we are counting {AA} if {A} appears twice. Similarly {BB}, {CC}, or even {**{}{}**} should we want to count all of the combinations of nulls. In our case, counting the number of nulls, where a term in a set, e.g. {A}, appears twice will be redundant, more on that in the next section. At level q=3, we are performing the same type of binary combination. In this case, we are combining the results of level q=2 with the basic set again. We can see that the chance of finding {ABC} is one out of seven.

This can be represented in terms of bits, and breaks. •••│•••│•│ Each group of •│'s represents a possible accessible q-level. A set of terms with {A}, {AB} and nothing in q=3 would be written •│•││, where the double ‖ indicates a null combination is present. This set of sequences can ultimately be represented as a program. The complexity of that program can be represented by Kolmogorov's algorithmic complexity, which is essentially Shannon's (average) information entropy plus a constant.

Each of these subsets represents a possible way that a researcher may find this message. Often, as we know, we use only elements of some research, that is single or double or more sets of terms. Each of these are legitimate accessible states of the message. The higher q-level terms can be viewed as higher level concepts. You can see by inspection that it is not possible using this approach to have a q=3 term without filling the lower level q-level states. At some point the combination of a q=2 or q=3 set of terms can take on meaning as a primitive terms in and of itself. These higher q-level sets take on the meaning of a higher level of abstraction. They can then be considered representations of a concept, which may be replaced by new single terms.

At that point, it becomes a q=1 set. We would expect that the higher level q sets will exhaust when they become more and more frequently used. This seems to be consistent with the abstraction discussion (Whitehead 1910), and learning models (Newell 1980) (See 11. Abstraction, p91, and 10. Learning Curves, p90 and Chapter III). Shannon (1948) illustrated this using a telegraphy notation where, a birth or death was simply represented by a few terms. The receiver understood that those few terms implied that a baby boy was born on a certain date, and other appropriate details. We do the same thing when we learn. We follow an economy of symbols and the principle of least effort (Zipf 1949) discussed earlier.

We could look at the entire message of the publication (the article or report), and we could, in fact, look at every term in the publication and determine the frequency of occurrence of the set of sets of terms. If we were looking at every term in a message or report, we could also populate the lower half of the matrix. This would permit the determination that {AB} was different from {BA}, because {AB} has A preceding B and the reverse is true in the case of {BA}. This is how the analysis would be done for a free text study.

For the purposes of this experiment, we are using a bibliographic record, and only examining the descriptors. In a descriptor field, we would not expect the term to be entered more than once, and the order is not significant in the data source used in this study. We do assume that the terms in the descriptor field are representative of the topics covered in a message. Further, that the message terms in this field are symbolic of the

topic (technology) being discussed. This is reasonable, since this is the reason an organization like the Institute of Electrical and Electronic Engineers indexes material in this way. Similarly the Library of Congress, catalogs by index terms that are representative of the document. It would make no sense to go through the trouble of indexing material with the irrelevant terms when the mission is to make the knowledge available. The further assumption is that the frequency of occurrence of terms is representative of the attention the subject matter is getting.

Imagine a world where the language was represented by a data set with a 1000 records (messages). Of these 997 records contained only term {A} while the remaining records contained the terms {AB}, {AC} and {ABC} we would expect that this world was generally concerned with term {A}.   We can quantify this in a probability. The probability of finding the single term {A} is .997, while finding {ABC} is .001, or a one in a thousand chance – not too likely, but possible. We can calibrate an expectation of finding a term or particular combination of terms for these probabilities. This ability to quantify the probability of finding terms, predicting the future based on the present expectations from distributions is key to the approach used in this research.

The context gives us a degree of insight into the likelihood that there will be more of the same to follow. Building from the needs of cryptography, this is where the fundamental work of Shannon (1948) started. He illuminated the way to make decisions based on these probabilities, using the concept of entropy. One can even represent the number of decisions needed for absolute certainty. This research launches from this point.

B. TECHNOLOGY TRANSITION: ANNOTATED BIBLIOGRAPHY

There are many studies that have driven down into the implementation details of technology diffusion and infusion. This section presents a survey of the relevant technology transition literature that supports the development of the model presented in this dissertation. This section also provides a link to the implementation studies that are available to date, so that the model can benefit from organizational and technology β's unique to a local study.

The appendix of this dissertation contains an annotated bibliography in two parts. The first includes the basic work done by the SEI (Przybylinski 1988). The second part resulting from this dissertation research is the addition to that work and brings it up to date with a large number of newer citations.

Many of these have been annotated with an abstract. In many cases, e.g. SEI edited proceedings of the International Federation of Information Processing Technical Committee 8 (TC8) Working Conference on Diffusion and Implementation of Information Technology (Levine 1994), the annotated bibliography of each of the key papers is included. With one exception, the key software technology transfer research papers referenced in this section include the papers cited in those papers. This provides an excellent starting point for future research. The exception is Rogers 1983, 1995 which has twenty four (24) pages of references (2700 references by Przybylinski's count). These references of Rogers represent the most important work on the broad topic "diffusion of information", according to Rogers (Rogers 1983, p. 414). The SEI will soon be publishing this annotated bibliography, which includes the sites to the material that each bibliographic citation references (Saboe 2001b).

C. STATISTICAL ELEMENTS OF THE TECHNOLOGY TRANSITION MODELS

This section covers the definition of terms as used in this research. The use of terms and aspects that factor into the development of the proposed technology transition

models is developed. The historical basis for the use and the thread of connection to the current research and between probability, information, and uncertainty is described. A general discussion of these terms in the context of information-communication theory is presented. The notion of entropy both in information theory, statistical mechanics and as used in thermodynamics is introduced. We discuss the stochastic models and the relationship to a dynamical system model. Elements introduced here are known in the literature, and are accepted with out the need to prove them. This section sets a context and a point of departure for Chapter III.

1.   Probability

What is this technology transfer engine? Is it deterministic? Is it probabilistic? While it may appear that we could have non-determinism here, this is not the case. If we could know all of inputs, there is a deterministic relationship, however, it impossible to know all of the inputs. So, we must distinguish between non-deterministic and probabilistic. We simple don't have enough information to accurately predict the result. This is because there are uncertainties in *input* to the technology transition process. There is a spectrum of distributed inputs, and this feeds a deterministic flow of correlations ordered in time. These are due to deterministic interactions, which yields a result.

As indicated before, we have a irreversible flow of correlations that are ordered in time just as there is a flow of communication in society. This leads to an equilibrium solution if we have a technology that is stabilizing. There is a distribution of the input variables at work, all with probabilities attached. This affects the likelihood of discovering, extending and refining a technology, re-transmitting the technology and acceptance of a technology. We are dealing with probability, uncertainty and risk. While risk can be defined as the product of the probability of an event and cost of the event, for our purposes, we deal with uncertainty and risk as the same thing. We ignore the cost component in this development. In a real program office, the cost elements can be later added in to perform trades and risk assessments. It does not matter, whether an "objective" classification is or is not possible. We deal with a "subjective" probability concept. (Hirshliefer 1992, p. 10, and Savage 1954).

2.   Information, Uncertainty

Information is a difference in matter-energy [change of status – i.e. state] that affects the uncertainty in a situation where a choice exists among a set of alternatives (Rogers Kincaid 1981).   "Information is something which reduces uncertainty. Communication is exchange of information." (Wiio 1980, p. 18)   Information is the ability to choose between alternatives reliably. Before you send me an email, I cannot reliably, guess your message. After I receive it, I can do so. I have gained information (www.aip.org).

Uncertainty is the degree to which a number of alternatives, the multiplicity of options, are perceived with respect to the occurrence of the event and the relative probability of the outcomes. Uncertainty implies a lack of predictability, of structure and /or information.   This multiplicity of option states can be quantified in terms of *entropy*.

Entropy and uncertainty can be considered synonymous (Jaynes 1957). Jaynes made the linkage between statistical mechanics as we know it from (Gibbs 1903), and entropy as we know it is thermodynamics, by relating a common concept to both – maximum entropy. Mathematically, maximum entropy has the important property that no possibility is ignored. It assigns positive weight to every possible situation that is not absolutely excluded from the information.   It is the state where we can deal with equilibrium properties. According to Jaynes, this is quite similar to an ergodic property.

The macro equilibrium state of a system (this is what we see in classical thermodynamics), is the macro equilibrium entropy, *S*. From Boltzmann, we get

$$S = kP \ (\{p_i\}) \tag{2.2}$$

This is when the maximum value P of the statistical entropy functional $P(\{p_i\})$ through the Boltzmann constant[7] k. Where $P(\{p_i\})= \ln \Omega$ is the uncertainty. Where k for {nats, bits, bytes, or Joules/ ° Kelvin} is $\{1, \dfrac{1}{\ln 2}, \dfrac{1}{\ln 256}, 1.38X10^{-23}\}$ respectively. We can convert the natural log, ln, to $\log_2$ easily.

$$\log_2 x = \frac{\ln x}{\ln 2} \tag{2.3}$$

The probability distribution $\{p\}$ is on the set of available microstates $\Omega=\{i\}$ or multiplicity. The functional $S=kP(\{p_i\})$ needs to satisfy two general properties. (i) P must be positive, taking the value zero only in the case of absolute certainty ($p_i = 0$ for all states, except for a given state j for which $p_i = 1$). (ii) P must increase monotonically with increasing uncertainty. In addition, a third condition is required. (iii) The P is additive for independent sources of uncertainty (Bayes 1763), (Planes 2002). Because of this, we have the property of extensibility. This means if you add or subtract these quantities which contribute to uncertainty, the system size – the extent -- changes. Adding these quantities requires a product of the probabilities.

We can compose a system like this, with a system composed of two subsystems which are independent, A and B, so that the set of microstates is $\Omega_{A+B} = \Omega_A \times \Omega_B$. Each microstate (i,j) can be specified by fixing a state $i \in \Omega_A$ of subsystem A and a state $j \in \Omega_B$ of subsystem B. If a probability density, $p_{(i,j)}^{A+B} = p_i^A p_j^B$, then $P^{A+B} = P^A + P^B$. (Planes 2002), (Munster 1969).

$$P \ (\{p_i\}) = -\sum_{i \in \Omega} p_i \log_2 (p_i) \tag{2.4}$$

3. Extensive and Intensive Properties

Extensive properties in the physical world are volume, mass, particles, energy, money, messages, records, etc. *Intensive* properties (e.g. pressure and temperature) on

---

[7] Shannon (1948) quickly points out that k is just a convenient constant to relate to our physical world.

the other hand are independent of the size of the system. A method to determine whether a property is extensive or intensive is to divide the system into two equal parts with a partition. Each part will have the same value for the intensive properties, but half for the extensive properties.   Examples of extensive and intensive properties are given in Figure II-11.

Extensive and Intensive Properties
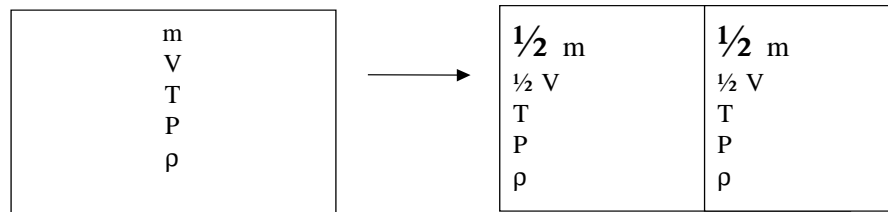
Extensive changes with the extent or size of the system

Intensive properties are not affected by the system size

Examples:
Extensive:mass, volume, energy, money, messages
Intensive: temperature, pressure

Figure II-11 Extensive and Intensive properties

It would be valuable to identify analogous extensive and intensive properties in the technology transition model, or in or general terms.

Property	*Ext-ensive*	*Int-ensive*	Thermodynamics Physical	Tech Transfer/ Information Communication System
Particle Mass	X		• *N* particles per mole	• Unit of entities, e.g. Term per some standard message length
Volume	X		• L^3 *(length^3)* or • *AL (Area * length)*	• *v * s* nodes consisting of authors * state change
Energy	X		• eV, Joules, BTU's	• Some conserved property • Messages, terms
Temperature		X	• °K degrees kelvin	• Some measure of change is cardinal, related to two variables *ext* and or *int*
Entropy		X	• S≥0 • $S=kP(\{p_i\})$ • $S = k\ ln\ W$ • Always increases • Additive for Independent Identical Distributions	• Similarly defined for information (Shannon 1948) • $S=kP(\{p_i\})$ • $S=-\Sigma p_i \log_2 p_i$ • Maximum entropy – uniformly distributed probabilities, same as thermodynamics
Pressure		X	• Force per Area	• Messages per node
Density		X	• Extensive property per volume	• Messages per *v* nodes (sum of *v* authors)

Table II-6 Property Relationships

Particles are analogous to sets of terms in a message. A message is made up of sets of terms. Counting all of the sets of terms is the same as determining the number of entities, particles. Just like in molecules, some entities have more weight than others. If all null and single term sets have the same weight, the analogy is a set of sets of terms e.g. {}, {A}, {B}, {C}, {AB}, {AC}, {BC}, {ABC}. {A} is "lighter" than {AC} which

is a composite of two if a term is made up of {A}+{C}. There should be some relationship between changing the status of a term and analogous principles in the physical world. e. g. Newton's laws (see the next section).

Volume in the physical world, is in three dimensions measured in some length units. We can get a volume with units of l^3 by measuring the volume. Integration over small dl is used in continuous space. For a discrete system, we count the points defined in phase space. For the models, this volume is defined in only two dimensions, nodes (a publisher) and state points. This is discussed in further detail in Chapter III, on page 444.

In a classical thermodynamics model, energy is measured in Joules, or BTU. It is often convenient to measure energy units in electron volts, which is the kinetic energy of an electron that has been accelerated through a voltage difference of one volt. This is moving an electron from its status at point A to point B.   This is directly related to the conservation principle, the 1st law of thermodynamics, and Newton's 3rd law. The first law of thermodynamics says that *energy is conserved and transformed*. Energy is a primitive and essential thermodynamic function.   It is a mathematical abstraction. (Abbott 1989, p1). Newton's 2nd and 3rd laws similarly constructed using the principle of conservation.

Law 1 *"Every body preserves in its state of being at rest or moving uniformly straight forward except insofar as it compelled to change its state by forces impressed."*

Law 2 *"A change in motion is proportional to the motive force impressed and takes place along a straight line in which a force is impressed."*

Law 3 *"To any action [change of state] there is always an opposite and equal reaction; in other words, the actions of two bodies upon each other are always equal and always opposite in direction"*[8]. (Newton 1726, p417).

Newton says in definition 3 of law 1, "because of inertia of matter, it is only with difficulty put out of its state either of resting or of moving." In Newton's interleaved copy of edition 2, he adds the following which was never printed: "I do not mean

---

[8] This is the exact statement taken from Newton's original work. Modern texts have often changed the wording slightly on each of his laws, but the original statements give us the closer intent of the law to this

Kepler's force of inertia, by which bodies are moved toward rest, but a force of remaining in the same state either of resting or moving." (Newton 1726 p404). Change of state, or status, must overcome some inertia. E.g. changing v_0 to v_1 meaning to change from an initial state, say a velocity, to a new velocity. Even to change one orientation of one atom, or one bit, such a change of state, takes some force or stimulus. Something must happen to change the state of information otherwise it stays in its current state.

In Figure II-12 below, we show the relationship using a Venn diagram, that shows the probability of two sets can represent this conservation through correlations of extensive properties at the intersection consisting of mutual information. The left hand subsystem *A* is composed of the sum of the uncorrelated part P (*A/B*), plus the correlated part *I(A;B)* still equal to the total and the P (*A*), where *I(A;B)* is the shared mutual information. This is the equal and opposite amount required by the 2nd and 3rd laws of Newton. Similarly, the right hand subsystem *B* is composed of the sum of the uncorrelated part P (*B/A*), plus the correlated part *I(A;B)* which is still equal to the total and the P (*B*). Looking at relation 4, *I(A;B)=I(B;A)* and other relations in Figure II-12, we see how the conservation principle is realized. The key is not conservation of energy in this research, but rather the conservation of the correlated components of extensive properties in two interacting subsystems (Planes 2002). What one subset loses, the other gains.

Mutual Information and Conservation of Extensive Properties

$$P^{A+B} = P^A + P^B \qquad (1)$$
$$I(A;B) = P(B) - P(B|A) \qquad (2)$$
$$I(A;B) = P(A) + P(B) - P(A,B) \qquad (3)$$
$$I(A;B) = I(B;A) \qquad (4)$$
$$I(A;A) = P(A) \qquad (5)$$

$P(A,B)$

$P(A|B)$ $I(A;B)$ $P(B|A)$

$P(A)$ $P(B)$
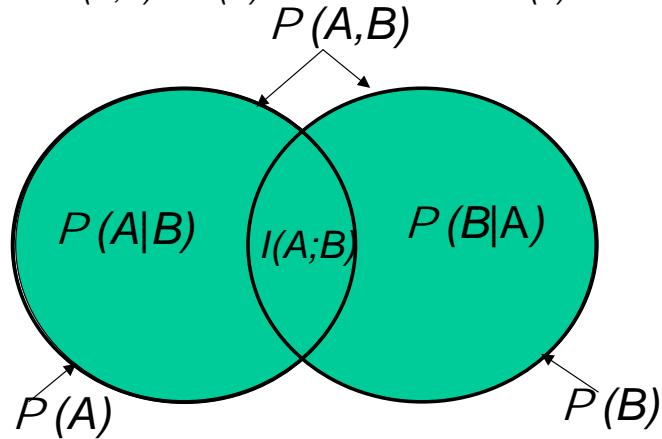
Figure II-12 Mutual Information and Conservation of Extensive Properties

4.   System, Control Volume, System Boundaries, States

The research refers to system, control volume, system boundary, states, in the usual ways. The application of the conservation principle requires a system and surroundings defined as a discrete portion of the universe. A system is any object, any quantity of matter, any region of space, etc. selected for study and set apart (mentally) from everything else, which then becomes the surroundings. The systems we are interested in are finite. There are two points of view, macroscopic and microscopic. Macroscopic takes into account the coarse characteristics of the system with intensive properties regarded as state space coordinates for example a *T-S* (temperature-entropy) diagram shown in Figure II-13, shows a third intensive variable *P*, pressure. Figure II-14 shows a typical *P-V* (pressure-volume) diagram for the same cycle.

In thermodynamics, there are the concepts of *Q,* heat, and companion quantities, *W,* and *H,* mechanical work and enthalpy, which are convenient mathematical concepts respectively. These are related to an internal energy *U*. *U* is a function of the internal, microstates discussed before. $\Delta U=Q-W$. The change in the internal energy ΔU is the difference between the energy put in as heat *Q* (some stimuli input), and *W* useful wok out[9]. In our model, we are stimulating researchers and developers to produce messages that are used. Those that are generated, but not used are wasted. This is related to the system efficiency[10] η.

In differential form, $\Delta U=Q-W$ is written

$$dU = \delta Q - \delta W \qquad\qquad (2.5)$$

All energy exchange with the surroundings, in this case, serves to just change the internal energy. If in addition the process is adiabatic i.e. no heat transfer with the surroundings), then *Q=0,* and this becomes

---

[9] We have left out physical energy terms relating to the physical analogs for kinetic and potential energy at the system level. Even in thermodynamic analysis, most common problems do not need these energy quantities.

[10] The word efficiency comes from the Latin "efficax"=effect. In mechanical efficiency, all we are interested in is the effect, "work". In every other kind of efficiency, we take the ratio of $\Delta E(x)$, "energy change" actually used to obtain the effect *x* to free "energy" ΔF, released (applied) to obtain the effect. $\eta(x)= \Delta E(x)/ \Delta F$. or $\eta(x)= specified\_output\_"energy"\_change / input\_"energy"change$.

$$dU = -\delta W \quad \text{(adiabatic)} \qquad (2.6)$$

This says that for a system changed adiabaticly from one equilibrium state to another, the work should be independent of path. Or ΔU should depend only on the end states. This research will explore that relationship in an example in Chapter III.

In the case of this research, the relationship between work and internal free "energy" states is dealt with as a potential. In statistical mechanics, thermodynamics and technology transfer dynamics, there is a potential that is the difference between the macro state when the system is at equilibrium, and the current state of the system. We see this because in statistical mechanics, the macroscopic property view is based on microscopic principles. E.g. equal probability of microstates gives the macroscopic description.

This potential is realized in a manner similar to the general Massieu-Planck generalized ensemble potential functions (Munster 1969 Chapter III) and (Planes 2002). We hinge on Jaynes (1957) relationship that linked Gibbs thermodynamics and statistical mechanics to information theory. Accepting that, we can have available to us the Gibbs postulate that the quantities calculated by thermodynamics are identical to those calculated by statistical mechanics. In our case, we can indicate the probability that a term is in state j as $P_j=N_j/N$, where N_j is the number of terms in state j, and N is the total number of terms in the system. Similarly we can determine the distribution function for velocity $P(v)$.

So, we can use the concept of ensemble potentials if we are careful about the conditions to define equilibrium.

These potential relations are independent of the conserved quantity, they simply relate a current state to an equilibrium state. From these relations, we readily see that the free energy $F=U-TS$. Where U is the internal organizational energy of the system, T and S are the temperature and entropy of the system. S represents a systems present organization. Only part of the system's total potential is locked up in the present organization. The rest of the accessible "energy" states are "free" from the current organizational constraint. Gibbs (1906) gave us this for physical systems. Maxwell tells us that

$$T \alpha <KE> \alpha <v^2> \qquad\qquad (2.7)$$

This says, the higher the absolute average velocity, the higher the temperature. In Chapter III, we will present a method for interpreting the velocity (rate of change of a state). This coupled with the partition function can get us to the proportionality constant of temperature.

State Space Diagram
Intensive Properties Temperature, Entropy
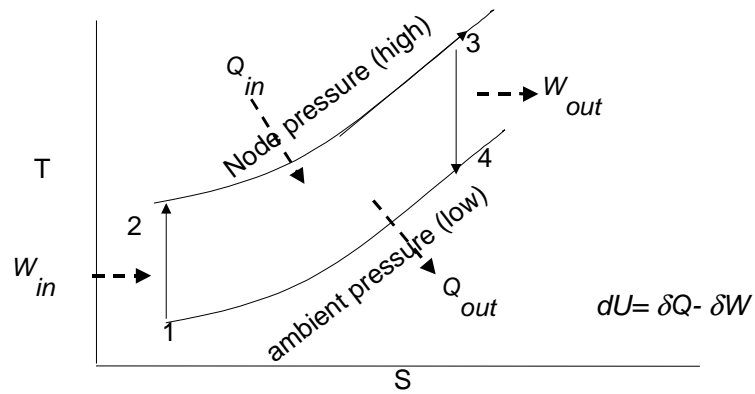
Figure II-13 Intensive Properties State Space Diagram

State Space Diagram
Extensive (V) Intensive Pressure (P), with
Isentrops (S)
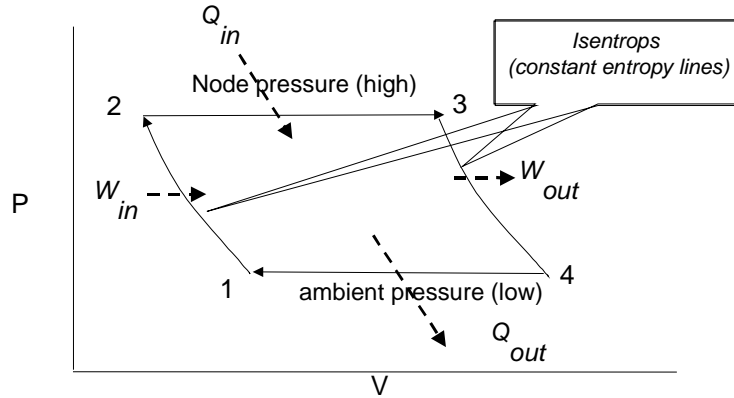
Figure II-14 State Space P-V (Pressure-Volume) Diagram

The microscopic view addresses the internal structure and details of the system in a series of canonical[11] decompositions. Microstates represent these internal structural details and properties. *U*, internal energy can be related to the multiplicity of microstates (Schroeder 2000), (Planes 2002), (Munster 1969). To specify the microstate Ω of a system you must specify the state of each individual entity. If we specify the state more generally, by saying how many are in a given state, we are referring to a macrostate. The number of microstates corresponding to a given macrostate is called the multiplicity of that macrostate (Schroeder 2000). For example, assume there are 100 types of coins (an alphabet of 100). The total number of microstates is 2^{100}, since each of the coins has two possible states. The total number of macrostates is only 101: 0 heads, 1 head, 2heads,… up to 100 heads. There are *N* coins (in this case 100 different of coins), the multiplicity of the macrostate with *n* heads is

---

[11] Canonical means broken down into finite primitive arrangements. It comes from religious heritage when the Catholic church laid down cannon law. Meaning the variable conform to a scheme that is both simple and clear.

$$\Omega(N,n) = \frac{N!}{n!(N-n)!} = \binom{N}{n} \text{ or } C_N^n \qquad (2.8)$$

The last expression is the standard abbreviation for the quantity of combinations C_N^n of n items chosen out of N. So if we have one each of coins A, B, C … 100 in this example we have an equal probability when all of the combinations exist once. If we have multiple (A, B and C) coins and only one of the rest of the types of coins, we will have a biased set of combinations which are possible, and the equilibrium is skewed. This research will show that we get a skewed distribution that is biased toward pairs and triple sets of terms (possible combinations of primitive message sets) take the form of Boltzmann's distribution. What is obvious is that it is VERY unlikely that we find combinations outside of the most likely states. (Schroeder 2000 Chapter 3), (Nash 1972, p12), (Castle 1965 p99). Figure II-15 shows the possibilities for an alphabet N of 128 single different types of coins. This is really a VERY, VERY tall skinny distribution. The confidence limits around the mean (the peak) is represented by $1/\sqrt{10^{36}} = +/-2x10^{-19}$. The number of microstates associated with each of the $(N+1)$ configurations is always calculable. While we can always calculate the number of microstates, and although always imaginable, as *feasible* in principle, we find that some trial and error possibilities are wholly impossible in any reasonable time, especially when the number of combinations is many orders of magnitude. Therefore, we have available to us the tools of differential calculus and we measure some experimental data. We see that the predominant number of configurations corresponds to the peak of the curve, where the tangent line must lie horizontally.
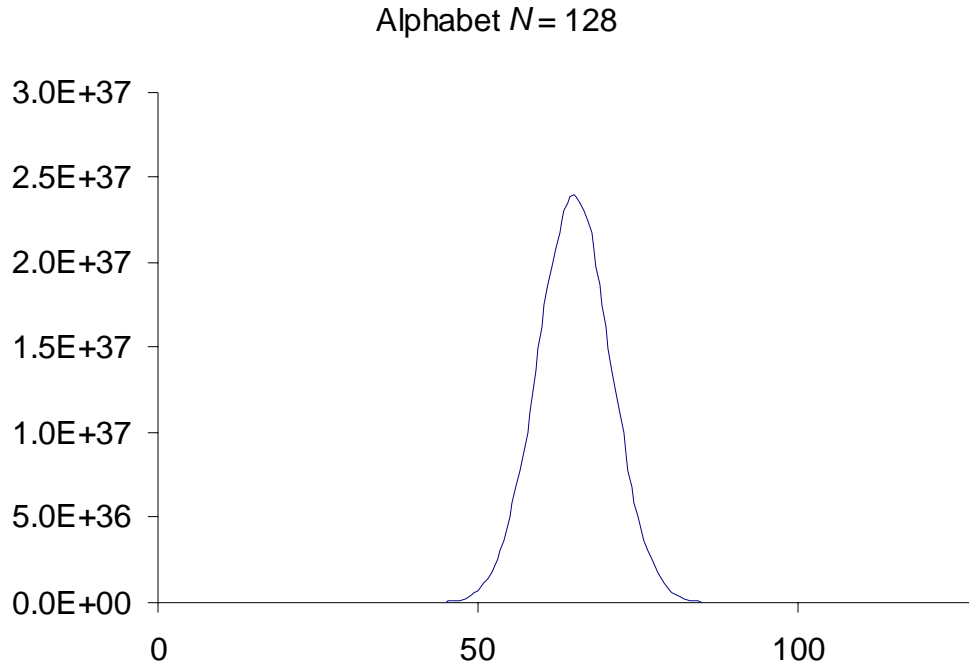
Figure II-15 Note distribution of configurations the *Y* axis is on the order of 10^{36}

So the criteria for the predominant configuration is simply that $d\Omega/dX = 0$, where dX denotes a change from the predominant configuration to another configuration only "infinitesimally" different from it. The change in $d\Omega$ and dX is not infinitesimal in the absolute case, but differential calculus demands that changes be infinitesimal in the *relative* case. This condition is met with even 10,000 or 100,000 units of multiplicity and only a dozen quanta of "energy" states. For a sufficiently large assembly, we can regard Ω as an effectively continuous function of the configuration index X which we refer to as q-levels in this research. So we need not be reluctant to use the criteria that $d\Omega/dX=0$ to identify the predominant configuration. This follows from any development of quantum statistical mechanics (Schroeder 2000), (Nash 1972), (Castle 1965).

There is a fundamental assumption in statistical mechanics that in an *isolated* system, all accessible microstates are equally probable. When two systems are contact, for example, a system and the surroundings, we are equally likely to find the combined system in any of its accessible microstates. So, we can always compare a distribution

with the maximum entropy, minimum extensive property distributions using relative entropy. It turns out that *at equilibrium, the configuration of an isolated macroscopic system ensemble is typically that described by the Boltzmann distribution laws.* (Nash 1972 p25). If we are seeing a Boltzmann distribution and a number of criteria are met, we can estimate the probability of choosing *n* items out of *N*. Since there are many possible distributions, to know which ones are right in our case, we measure them. We count the message subsets of *N and N_j* in our subsystems and super system. Then we also satisfy two conditions.

1) The number of messages *N* in the super-system consisting of subsystem *A* and subsystem *B* is constant. $N = \sum_j N_j$

2) The "energy" states of the super-system is constant $E = \sum_j E_j N_j$

Where $E_j N_j$ is the "energy" state of the jth level, in the canonical ensemble, we can let $\bar{E}$ be calculated from the statistical mechanics $\bar{E} = \sum_j p_j E_j$. Then we get $E = N\bar{E}$

The 0th law of thermodynamics permits comparison of two systems if they are in equilibrium with each other. Imagine the two systems are a subsystem and a reservoir. This arrangement is essentially, how a thermometer works. When the subsystem comes into equilibrium with the reservoir via energy exchange, the controlling variable on the mean "energy" states is *T*, the temperature. This is the resulting equation form Helmholtz Free Energy *F=U-TS* with the logarithm of the partition function Ω_c for the case of a Canonical ensemble.

$$F(T,V,N) = -kT \ln \Omega_C \qquad (2.9)$$

$$\Omega_C = \sum_{i \in \Omega_{V,N}} e^{-\frac{E_i}{kT}} \qquad (2.10)$$

where the available microstates are fixed in *V*, volume (i.e. the number of nodes), and *N* (i.e. the number messages built from the number, *n,* of terms – primitive messages). In a macroscopic view, we allow the exchange of energy by fixing the mean

energy and mean number of messages. In these ensembles, the determinations of the intensive and extensive variables are usually taken as natural variables of energetic potential. This is different, and contrasts, to the microcanonical ensemble, where the entropy is taken as the relevant potential. Therefore the basic equations of such ensembles are different and the equations above for the average values and fluctuations of the average values are also different (Planes 2002).

5.   State Equations

There are two types of problems, we would like to be able to solve. The first deals with *processes*, and the equations used deals with properties relating property changes of a system and the quantity of a conserved quantity (e.g. energy, mass, money, messages, etc) transferred between a system and its surroundings. The second is in the elucidation of relationships among the equilibrium *properties* of a system. We can derive these relationships by isolating the flows (heat, work, etc) dealing with reversible processes, and we can derive general relationships among equilibrium properties. These are no longer limited to the special kind of process initially used for the derivation. The properties are called *state functions*. (Abbott 1989 p59).

6.   Stochastic Model and Markov Chains

There were early efforts to say something about uncertainty over long sequences of words, word pairs and phrasings (Shannon 1948), (Mandelbrot 1953). We can assume that the terms used in the technology development and as published represents an analog to a piece of continuous prose, which is being written. Consider a book that is being written, and that it has reached a length of k words. We can designate the number of *different* words (later we refer to these as terms) that have occurred exactly i times in the first k words as $f(i,k)$, or in the notation of equation (2.8) C_k^i. That is, if there are 407 words that occurred exactly once each, then $f(1,k) =407$. We have an assumption that the probability that the $(k+1)$-st word is a word that has already appeared exactly i times is proportional to $i\ f(i,k)$, that is the total number of occurrences of all the words that

appeared exactly *i* times (Simon 1955). Simon also addresses the addition of terms and the disuse of terms. To this researcher's knowledge, he makes the connection to a stochastic process for the first time in the literature. Using terms as symbols, representing technology, we see that a there is a weaker assumption than the probability of a *particular* word occurs next would be proportional to the number of previous occurrence. Also, as Simon and Shannon (Shannon 1948) did, we can make an assumption that there is a constant probability that the $(k+i)^{-st}$ word be a new word – a word that has not occurred in the first *k* words. This describes a stochastic process in which the probability that a particular word as the one to be written depends on the words that have been written previously. This is fine if the number of words in the vocabulary is roughly constant or the rate of change in the terms being added or dying in a language is not significant. In English, for example, this birth/death of terms is small relative to the language.

For this technology transition study, we don't expect the terms relating to a technology to be constant. There will be new words added and some will die. Simon worked through this by assuming if one representative of a particular term is dropped, then all of the representatives of the term are dropped. He also made the assumption that the probability that the next term that is dropped will be equal to the probability of one with exactly the same number of representatives of one with the same relative frequencies (Simon 1955).

This result proves satisfactory for language analysis, and we now have a stationary condition to enable use of a chain of transitions. However, it does not quite work for a limited vocabulary, artificial language, as is seen in technology transition. It will be useful to define a model, which conserves a quantity, i.e. as a property decreases there is a change in another component, which increases. For example when two masses collide, there is a correlation of velocity, one increases, the other decreases until there is a mutual correlation of the shared quantity (in this case energy which is a function of velocity, hence velocity). We shall see that this mutual information and the conditional probability of messages and terms give us the quantity that enables the conservation principle for the studied models.

The chain of transitions is useful however. Stochastic processes of this type are known mathematically as Markov processes, and are extensively studied. In a Markov process the future evolution of a state depends only on the present state. There is a group of Markov properties of significance to information and communication theory. These are the egrodic (see 2. Ergodic Process, p273) processes, which simply stated says that every sequence produced by the process has the same statistical properties. So the letter, word, term or phrase frequencies obtained from particular sequences, will approach definite limits as the length increases independent of the particular sequence. The ergodic property means statistical homogeneity (Shannon 1948). The limits, provided by ergodicity permits us to establish a maximum, a reference datum that can be compared to.

In the study of technology, we would expect that a researcher, or publisher of a message, will use a process of associations using terms they have previously been written by sampling earlier segments of the term sequences they previously wrote. We would also expect that, there is a process of imitation, that is, sampling segments of terms from other researchers, and from terms heard.

Consider that the lens we put on the technology yields terms in a slice, of a length, of the entire sequence of terms in the technology's artificial language. We can deal with this as a control volume. A control volume, establishes boundaries, here it is a slice of the language, that represent the system under study. There will be further discussion of the control volume in Chapter III (see p211). What is required, and addressed, in this dissertation is a way to address the addition of terms across the control volume boundaries, and mixing within the control volume. Mandelbrot (Mandelbrot 1953) gave us the first hint of what will lead to a dynamical solution.

7.   Information-Communications Theory, Statistical Mechanics

Insightful developments in information-communication theory (Shannon 1948 and Jaynes 1957, 1957a) help bring together statistical mechanics used in physical systems typically used to "describe the dice" (i.e. the physical description) and "taking the best guess" (the gambling theory part). Miller (Miller 1956), Zipf (Zipf 1949), and

Simon (Simon 1955) tie together information theory, learning and skewed distribution respectively. For example, the 1st through 3rd laws of thermodynamics "help describe the dice", while the zeroth law, as well as Boltzman's (canonical) and Gibbs (grand canonical) (Gibbs factors are dice independent tools of statistical inference) (Fraundorf 2000) (Schroeder 2000).

The 1st law is the principle of conservation of energy. This deals with the quantity to be conserved.

The 2nd law deals with entropy. It says that the entropy change of any system and its surroundings, considered together, is positive and approaches zero for any process which approaches reversibility. The second law addresses the quality of the property being conserved. It can also be shown that the spontaneous flow of a conserved quantity *stops* when it is at or very near its most likely microstate, that is the maximum entropy state (Schroeder 2000 p59), or in equilibrium with another system. The second law can also be viewed as a very strong statement about probabilities.

While it may be initially troubling to the software community to have to think about physical properties (software has no physical properties, weight, temperature etc.), we can link the constructs of logical and physical space through *entropy.* Kolmogorov (Kolmogorov 1956, 1965) defined and showed various approaches to quantitative definition of information. Li (Li 1997) illustrated applications of Kolmogorov Complexity. Uspensky (Uspensky 1992) addresses the relationship of entropy and varieties of Kolmogorov's complexity[12]. Farmer (1983) showed the relationships of dynamical systems, information measures and dimensions and entropy. Prigogine links irreversibility, dynamical systems and entropy in distributions as inputs, verses single point trajectories. If we can take advantage of this body of knowledge we, as software

---

[12] Kolmogorov's complexity is related to Shannon's entropy, and the notion of randomness. The main idea was developed by Bernoulli in 1713 where he stated that an experiment (recognize that this is what we do in technology transfer or evolutionary development) with probability of success *p* is repeated *n* times, then the proportion of successful outcomes will approach *p* for large numbers (Li 1993, p. 55). Bayes put particular definition on the term probability as the *measure* that an expectation depending of the truth of any *past* fact or the happening of any *future* event so that the more valuable as the fact is likely to be true, or the event is more likely to happen (Bayes 1763 Barnard 1958, p. 298). He also suggested the "inverse of Bernoulli's problem". Laplace (Li 1993, p. 46), further analyzed the *inverse probability* as is is also known and referenced Bayes in his discussion but this could not could not be developed by Laplace at the time.

engineers are provided huge leverage in lexicon, theory, and analysis. This ultimately provides the potential for accelerating software technology transfer and keeping the evolutionary development process under intellectual control.

8. Quantitative Zeroth Law

Let's assume a model of the quantitative version of the zeroth law cited above. Here is a theorem of statistical inference not involving energy at all. It applies also to thermally unequilibrated systems sharing other conserved quantities provided the only prior information we have is how the multiplicity of ways that a quantity can be distributed depends on the amount of that conserved quantity to begin with. Since this abstraction relationship relies only on the probabilities of the encompassing state property, we have a property that depends on the conserved quantity.

9. Entropy

Entropy as a concept can readily be seen as logical entropy (think of it as a measure of uncertainty, noise, non-signal, process inefficiencies, the percentage of work resulting in defects and requiring rework, etc) and physical or thermodynamic entropy (i.e. mixed-up-ed-ness, disorder, disorganization, etc), which is the quantity of energy not available to do work. Logical entropy is Shannon's entropy (S_H) as defined by Shannon on his treatise on communication theory (Shannon 1948). Shannon's theory says that the entropy of an information source measures how well its behavior (e.g. the next symbol in a sequence it produced) can be predicted.

Mixing entropy can be represented by the eigenvalue of a bakers' transformation function. This baker transformation in state space represents entropy in terms of folding, stretching, translation and rotation (Spiegel 1998 p292). This transformation is the representation of a *dissipative structure.* These are structures with an innate capacity to dissipate anything that comes in to disturb the system. The term "dissipate" is somewhat unfortunate, because what really occurs is *integration* not *dissipation* (O'Murchu 1997 p.168). The entropy is the quantity of information not available to help

us work, yet is valuable to understand if the objective is propagation and diffusion. The relationships are developed in Chapter III.

Recently a number of undergraduate texts are illustrating entropy as the accessible state multiplicity for quantities that must be conserved -- e.g. volume, and particles. The notion of conservation of a quantity is important to this research, as this could be momentum or more importantly information. This is understood from the logical-mathematical interpretation of the equations vs. physical interpretations. It requires us to step back and look at conserved quantities in the mathematical sense, then map those to our problem. Further, entropy, temperature or coldness (*1/T*) and heat capacity have been developed on the basis of information units alone (Fraundorf 2000).

10. Learning Curves

We can associate efficiency with how well we automate the process of acquiring knowledge. Learning provides leverage and yields efficiency. When we get efficient, we free up cognitive capacity, which in turn permits future learning. A large number of papers have examined, and reviewed the notion of a learning curve. As early as 1919 Thurstone (Thurstone 1919) considered logistic, exponential and hyperbolic functions. The log-log form was dismissed by Mazur and Haste (Mazur 1978), but Newell and Rosenbloom did extensive analysis and examined the theoretical basis of the power law. They showed that power law learning is like exponential learning when examined in terms of the local rate of learning. Newell and Rosenbloom (Newell 1981 p2) state "There exists a ubiquitous quantitative law of practice. It appears to follow a [what they call] a *power law*; that is plotting the logarithm of time to perform a task against the logarithm of the number of trials always yields a straight line more or less." They refer to this as the *log-log linear learning law* or the *power law of practice*. They also developed a form of the power law to deal with spans of patterns, which appears to take a form that may be very relevant to follow-on research. This *chunking* form of the power law learning is suggestive. There could be a relationship to the models developed in this text.

11.   Abstraction

The entropy discussion so far only gives us a logical-mathematical tool set and framework. There are some other aspects that need to be addressed. One that is still floating around from Plato's vignette about Meno's learning is the notion loosely referred to as "unpacking". Can this be tied back to entropy and communications as well? This seems to suggest a terms-of-reference and an abstraction requirement to minimize the effort related to understanding the"encryption" and protocol needed to communicate the ideas in this research.

The user simply needs to know how to use the product. i.e. product-use and process-use knowledge. For example, the general population only needs to know how to start a car, drive a car (after training), know the reason for fuel, fuel a car and observe faults. Concrete acts requiring little to no thinking to communicate messages require little additional processing steps, and hence the least uncertainty or opportunity to add noise to the signal. A way to look at this is to create a set of nodes representing states in a hierarchy. In some models these can be hierarchical or collector states.

This representation of states as nodes in the dimension of depth of knowledge is not new. In writing *Principia Mathematica*, Russell and Whitehead (Whitehead 1910) were forced to construct a hierarchy of types that would permit logical statements to refer to other logical statements. In their theory, a proposition could take the place of a variable if it were interpreted as being on a lower level than the meta-statement proposition. This relationship of logical hierarchical structures is very powerful here in terms of representing the depth of abstraction.

This is useful in the development of a similar approach that is adopted in order to apply the entropy concept. The application of the entropy notion based on hierarchical states permits use of the same units required for statistical inference techniques.

We now have access to a common dimension in the area of abstraction, uncertainty, and communication as well as temperature for the development of the theory and model for software technology transition, and evolutionary software development (Luqi 1989, Luqi 1991). It also leans in the direction required to represent software

applications (Berzins 1991). We have an entropy metric that works at a higher level of abstraction than is afforded by the system/engine/machine node interpretation. While we are counting every message, and structural combination in this research for experimental purposes, in actual practice we will be able to take samples. We no longer need to resort to counting every single particle or message or structure. Abstraction can also be useful when mapped to a scale to represent learning and competency for an individual or organization. Abstract representations in terms of combinations of terms at higher q-levels minimizes the effort required communicate. This means we can unpack a message easier and enables reliable and efficient processing of messages.

The hierarchy of types of technology transition, or evolutionary software development and/or software says:

For any selected state of a node, a lower level state diagram may be substituted.

The proposition implies that it is not necessary to know state at any level of the diagrams, but only their relative levels. This proposition requires that no state at any level is the same state as one on a higher or lower level. As with Russell and Whitehead's hierarchy, a state has only meaning in context.

The resulting axiom of reducibility to Whitehead's hierarchy is as follows:

The static relationships between states are not changed by the presence of sub-states.

In other words, the static probability of a state being active is not changed by the presence of its internal states. An important result is that the internal states have conditional probabilities, which rely on the probabilities of the encompassing state. (Grable 1994) While this research does not need to develop this further, all of the tools are available as a result of this research to do further analysis using conditional probability and Whitehead's hierarchy of states. In Chapter III, the impact of abstraction can be seen in amount of complexity in our representations results in increased understanding.

D.　　RELATION TO TECHNOLOGY TRANSFER

We can now start to see some of the elements that will constitute the technology transition model. It is clear we need to reflect the human in the process. The technology transfer literature is heavy with the focus on human learning. Uncertainty reduction is achieved through learning and the execution of informational activities (communications of a message of some sort) and the notion of irreversible combinatorial interactions and mixing (again in this context, a combination of input signals by some process and generating an output.) These are at a minimum, *probabilistic*, involving individuals that *reduce uncertainty* by performing an *informational* activity in the form of *learning*. *Chance* also plays a role. Fundamental to these ideas of learning and chance is *communication*. Both of these activities can be represented in terms of probabilities.

1.　　Leverage of Terms of Reference

The ability to bridge these two previously disconnected views of a physical and non-physical world conveniently provides powerful analytical tools to the software engineer. This is a nontrivial contribution to the software engineering community, we can put methods in the hands of software engineers that can be readily grasped by the mechanical, electrical, or communication engineer or anyone who has had some basic physics. This reduces the barriers to use by lowering the effort required to unpack, decipher and understand the protocol for the user community.

2.　　Software Technology Transfer and Evolutionary Development

This research makes an initial suggestion that software development, especially an unprecedented system development using an evolutionary, risk reductive approach, is very similar to the process of software technology transfer. This process is one of discovery, maturing thoughts on the application, fusing existing domain knowledge, and advancing the particular body of knowledge represented in the software product. The

body of knowledge advances when the prototypes, demo units and final product are delivered to the user community.

These two classes of processes, technology transfer and the evolutionary or spiral development model (MIL-STD-498 and Boehm 1988), are heavily laden with probability, and are primarily driven by external factors and the large proportion of human activity. We shall develop those points throughout this discourse on software technology transfer, and point out the analogs in the software development process, specifically in the case of evolutionary development.

The research suggests that these two cases are related. Similarly, the development of the theory will always keep an eye to an interesting challenge -- will the theory hold for software development and possibly to -- software itself. If this holds, we may very well have the first in-road into the development of what this researcher calls -- *Software Physics*. This research leaves until the end the speculation that software, a process itself, albeit a deterministic and predictable process, is similar in nature to the technology transfer and spiral development process with all of the uncertainty reduced or degenerated out of the framework. In the later sections of this discourse, these relationships will be more fully developed.

III.   METHOD AND MODEL

A.   METHOD AND MODEL DEVELOPMENT- FUNDAMENTALS

This chapter will review information theory fundamentals required to develop the various entropy models. A macro level basic entropy model is developed showing the trends of entropy S_H vs. time step k. Then a closed system consisting of two interacting subsystems is discussed. Here we show the relationship between extensive and intensive quantities. This permits developing a state equation relationship between properties.

A one dimensional state space representation in the form of a dynamical map is developed.   The data is related using the one dimensional dynamical equation $S_{H_{K+1}} = F(S_{H_K})$, where S_{H_K} is the input and $S_{H_{K+1}}$ is the output entropy at the macro level. The significance of stability, and the Lyapunov number for such a dynamical system is discussed. A two-dimensional finite difference map is introduced.

$$\begin{cases} S_{H_{k+1}} = F(S_{H_k}, N_{i_k}) \\ N_{i_{k+1}} = G(S_{H_k}, N_{i_k}) \end{cases} \qquad (3.1)$$

Where N_{i_k} is the number of messages at time step k. The subscript i is indicative of a performance band. Performance bands permit the partitioning of the community into groups of organizational nodes which possess statistically similar characteristics.   A possibility is to put all of the organization nodes and their associated authors that produce within +/- 1σ of the mean number of messages per time step together, and +2σ performing organizations together, and +3σ performing organizations together.   One could follow the development of Boltzmann and subdivide the population into ever decreasing size bins. We do, however, have to be careful not to reduce the bin size too small. If it is too small the statistical significance of the bin contents is lost and the probability distribution inside the bin will reduce to a single message trajectory.

Finally, a feedback model is introduced. Here a dynamical system of equations models introduction of new information, and the understanding of previously existing information. This is considered at the organizational node level of interaction. The eigenvalue of the feedback model also represents and entropy. We will see that a tuning parameter permits closely aligning the dynamical system model trajectory toward stability over time with the macro level information theoretic model. This tuning parameter might be viewed as relating to the learning rate.

B.   INFORMATION THEORY - SHANNON'S ENTROPY

Informally, information measurement can be understood as anything that increases the variance also increases the information. Generally, variance is usually stated in units of measure, e.g. meters, volts, etc. The amount of information is a dimensionless quantity. When we have a large variance, we are very ignorant about what is going to happen. If we are very ignorant, then when we make an observation, it gives us a lot of information. On the other hand, if the variance is small, we know in advance of our observation how the result is likely to come out; hence, we get little information from making the observation.

Shannon (Shannon 1948) best explained entropy in a theory that assigns a quantity of information to an ensemble of possible messages. All messages in the ensemble being equally probable, this quantity is the number of bits needed to count all possibilities. This says that each message in the ensemble can be communicated using this number of bits. However, it does not say anything about the number of bits needed to convey any message in the ensemble. So this approach can be reasonably related to a technology message. It could be simple and count as a message or as theory in a paper or demonstration.

Shannon is interested in the problem of communicating a *message* between a *sender* and *receiver* under the assumption that the universe of possible messages is known between the sender and receiver. (Li 1993, p. 61).

Technology maturation feels intuitively to be the stabilization of knowledge, based on prior information communicated in messages about a problem to solve. As with long run empirical evidence of dice throws, in gambling houses, or death statistics, in insurance companies, technology maturation similarly suggests that random frequencies, are *apparently convergent*. But it is clear that no empirical evidence can be given for the existence of a definite limit for the relative frequency. Yet the Bayesian approach quantifies the intuition that if the number of trials n is small then the inferred distribution (the future prediction) depends heavily on the prior distribution. However, if the number of trials is large, then irrespective of the prior distribution, the inferred probability condenses more and more around p.

Now suppose we have a technology we wish to implement - a problem to solve. If there is previously a lot of experience, then we either know exactly how to solve the problem, or we know the frequency of success for different possible methods. However, if the problem has never occurred before, or a limited number of times, the prior distribution is unknown or of limited value. Solomonff proposed a *universal* prior probability. The idea is that the universal probability serves as well as the *true* prior probability. In reality, we may not have a "prior" which is known for a technology. So we can define a start point as the probability that a fixed reference Turning machine outputs a sequence starting with x when the input is a fair toss of a coin (Li 1993, p. 58). In other words, we can start anywhere. Over a time, sequences and sets of sequences will develop. Almost all infinite strings (sets of sequences, i.e. messages) are irregular and satisfy all of the regularities of stochastic randomness.

Shannon does not capture the information content of the individual object (message) between a sender and receiver. He recognizes that "messages have *meaning* [… however …] the semantic aspects of communication are irrelevant to the engineering problem" of communication between a sender and receiver. (Shannon 1948) Kolmogorov's algorithmic complexity is a measure of the information content of the individual object (message). (Li 1993 p61) He shows that the complexity measure is related to the length of a message and prefix.

1. Entropy Review

The definition of entropy here is related to the definition of entropy in thermodynamics. Appendix A.   Information Theory, p274 provides a basic review of entropy in information theory after Shannon, Jaynes, Kolmogorov, Uspenski, and others as found in Li, (Li 1993) and Cover (Cover 1991). The basic entropy equations in this section, and the next three sections on maximum, joint, conditional and relative entropy follow closely to the development by Cover (Cover 1991). The basic probability relationships on which the entropy relations are built can be clearly seen in Bayes original work however (Bayes 1763).

Let X be a discrete random variable with alphabet Ξ and a probability mass function $p(x)=\Pr\{X=x\}$, $x \in \Xi$. $p(x)$ and $p(y)$ refer to two different random variables and are in fact two different probability mass functions $p_x(x)$ and $p_y(y)$. For the alphabet, with the given probability mass function, the definition of information entropy is:

$$S_H(X) = -\sum_{x \in \Xi} p(x) \log_2 p(x) \qquad (3.2)$$

S_H is the entropy measured in bits, and the log is base 2. $\log_2$ will be assumed throughout unless otherwise noted.

The base of the log is two for the natural units of information entropy as developed by Shannon (Shannon 1948). The entropy is a function of the distribution of X. It does not depend on the actual values taken by the random variable X, but only on the probabilities.

If $X \sim p(x)$ which means that the probability of use the random variable is representative of the element's usage over the alphabet, then the expected value E of a random variable $g(X)$ is denoted

$$E_{p(x)}g(X) = \sum_{x \in \Xi} g(x)p(x) \qquad (3.3)$$

The entropy of a plain random variable X can be interpreted as the expected value of $\log \dfrac{1}{p(X)}$, where X is drawn according to the probability mass function $p(x)$. Thus

$$E_{p(x)} \log \frac{1}{p(X)} = \sum \log \frac{1}{p(x)} p(x) = -\sum_{x \in \Xi} \log p(x) p(x) = S_H \qquad (3.4)$$

2. Maximum Entropy – Equal Probabilities

Here is an example. Let have a system where there are only two choices.

$$X = \begin{cases} 1 & \text{with probability } p \\ 0 & \text{with probability } 1-p \end{cases} \qquad (3.5)$$

then

$$S_H(X) = -p \log p - (1-p) \log(1-p) \equiv S_H(p) \qquad (3.6)$$

We see that $S_H = 1$ bit when $p=1/2$. Figure III-1 shows the basic properties of entropy. It is a concave function of the distribution and equals 0 when $p=0$ or 1. This makes sense because when $p=0$ or 1, the variable is not random and there is no uncertainty. The entropy is maximum when $p=.5$, which corresponds to the maximum value of the entropy.

Entropy vs Probability

Entropy vs. Probability

(plot with y-axis labeled "Entropy S_H", x-axis labeled "p(x) Probability")

<div>

Entropy S_H

$$S_H = - \sum p(x) \, log_2 \, p(x)$$

$$S_H = -(p) \, log \, p - (1-p) \, log \, (1-p)$$

Expected value

$$E_{p(x)} g(X) = \sum_{x \in \Xi} g(x) p(x)$$

$$E_{p(x)} log \frac{1}{p(X)} = S_H$$

</div>

Figure III-1 Entropy vs. Probability

Consider a system where input signals $X \in$ T. Specifically, where X is a set of terms,

$$\begin{cases} \tau \equiv \{term\} \\ 2^\tau = \{msg\} \end{cases} \qquad (3.7)$$

Where 2^τ is a set of all the subsets, often called the power set. Here is an example.

$\tau = \{A, B, C, D\}$

$$2^{\tau} \equiv \begin{cases} \{\}, \\ \{A\},\{B\},\{C\},\{D\}, \\ \{A,B\},\{A,C\},\{A,D\},\{B,C\},\{B,D\},\{C,D\}, \\ \{A,B,C\},\{A,B,D\},\{B,C,D\},\{A,C,D\}, \\ \{A,B,C,D\} \end{cases} \tag{3.8}$$

Now when the number of elements in $|\tau|$ =4, we get $2^{|\tau|} = 2^4 = 16$. Note also the distribution of sets. We have one null set, $\{\}$. We have four sets of singles. We have six sets of pairs. There are four sets of triples and finally one set of quadruples. Each of these are referred to as a q-level.

The maximum entropy occurs when we have an equal distribution of terms. So, for a message set where each subset of terms appears only once we define S_H as

$$S_{H_{\max}} = -\sum_{x \in 2^{\Xi}} \frac{1}{2^{|\tau|}} \log_2 \frac{1}{2^{\tau}} \tag{3.9}$$

The entropy maximum is at *1/p(X) or* $|\tau|$, or the number of sets of terms in the alphabet τ. In Figure III-2, we see the effect of sets of terms that are evenly distributed. In our model, we would *not* expect to see *.5< p(X) <1* as the result of integer number of sets of terms. This is because when we make decisions between two choices, one set of terms and another set of terms (an integer quantity), that yields a probability of .5. If we have one choice, one set of terms, we are certain of the answer, and the probability is 1/1 or by definition S_H=0.

Maximum Entropy

Entropy vs 1/ |τ| i.e.or p(X)

The entropy maximum is 1/*p(X),* or the number of terms in the power set.

- Entropy (bits)
- 1000
- 100
- 0
- 1
- 0.1
- 0

Probability of occurrence = p(X)

0   0.1   0.2   0.3   0.4   0.5   1
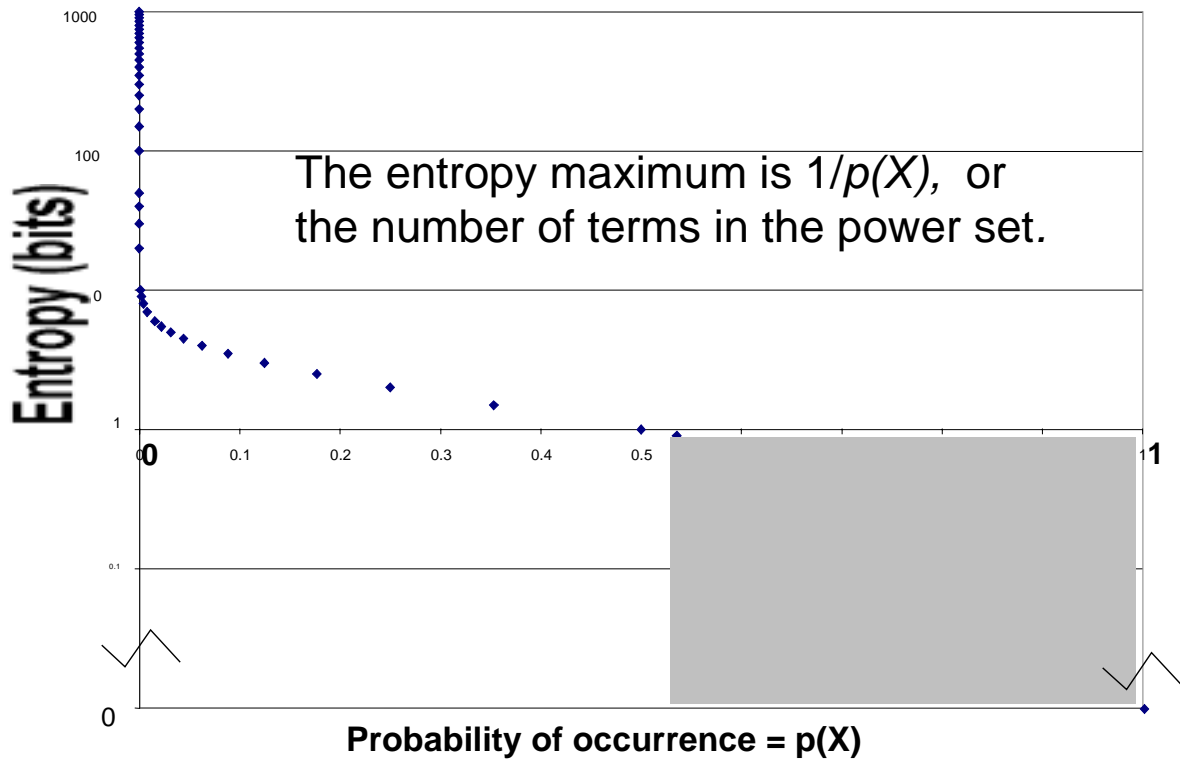
Figure III-2 Even distribution of terms, yields maximum entropy

The example vocabulary above, with an alphabet of |4| has a distribution of sets as seen in Figure III-3.
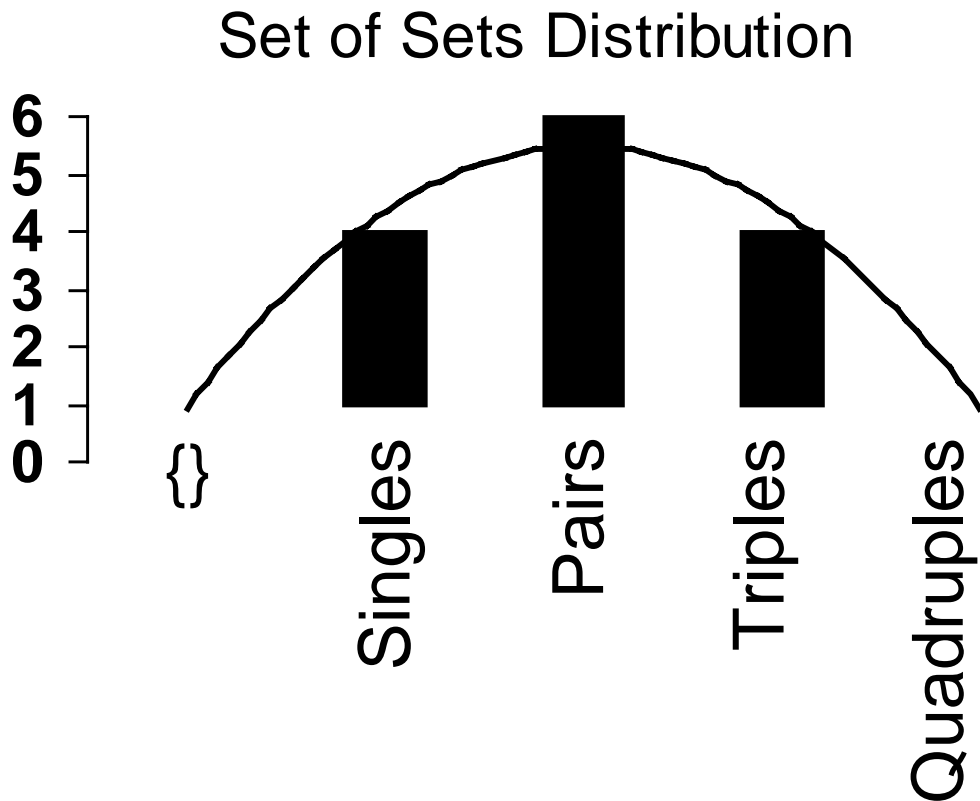
Figure III-3 Distribution of sets

Alphabets of |4| or |8| are tractable. Combinations available for |32| are already intractable. In Figure III-4 we have taken the log of the frequency plotted as a function of the combinations available in any q-level (sets of singles, doubles, triples, n-tuples). This illustrates how quickly the combination of sets grows, hence the probability of selecting a set is reduced.
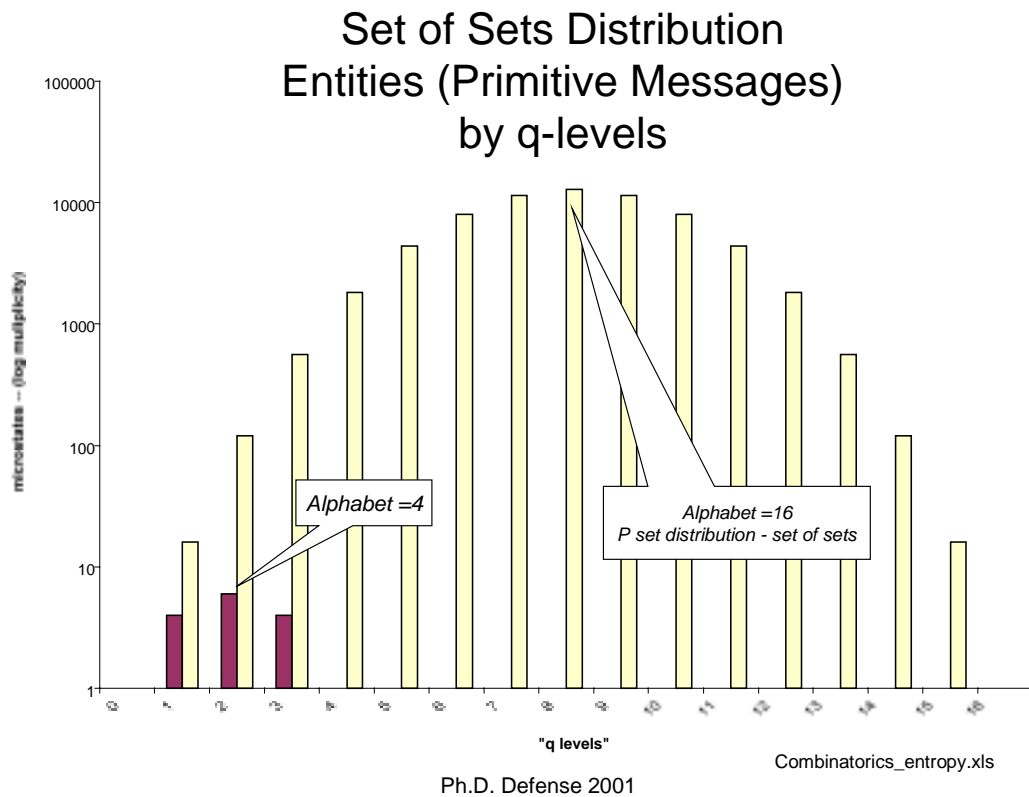
Figure III-4 Distribution of sets of sets (combinations) in an alphabet

It is appropriate to consider additional possible states that can occur. This would include pairs of terms, and triples, etc, until the sets of sets of terms are exhausted. Recognize q-levels are containing sets of subsets of |q| lengths. Their distribution indicates the most probable available states. q-level contents have distributions and different "weights"

e.g.	q level sets	distribution	"weight"
q=0	{}	1	0=0*1
q=1	{A}, {B}, {C},{D}	4	4=1*4
q=2	{{A} {B}}, {{A}{C}}, {{A}{D}}, {{B}{C}},{{B}{D}},{{C}{D}}	6	12=2*6
q=3	{{A}{B}{C}}, {{A}{B}{D}}, {{B}{C}{D}},{{C}{D}{A}}	4	12=3*4
q=4	{{A}{B}{C}{D}}	1	4=4*1

The "weight" of a set in $q_4 > q_1$ e.g. $\{\{A\}\{B\}\{C\}\{D\}\}_4 > \{A\}_1$. Weight of the level is product of the level, tells us how many terms were combined in a subset, and the number of sets in the level. We refer to Q_I weight of the q=i level. The weight of all of the levels summed is Q_c. Every one of these sets of sets is considered a message in our models. TO move a message from one q-level state to another requires some stimuli. We interpret this in the same way that Newton laid out his second law.

Distribution of Combinatorial sets of terms by q-level

Alphabet =32
q level number of sets of sets available in distribution

Ada_Affiliation (month)_entropyGraphsB.xls

Dec 2001   M Saboe   48
Ph.D. Defense 2001
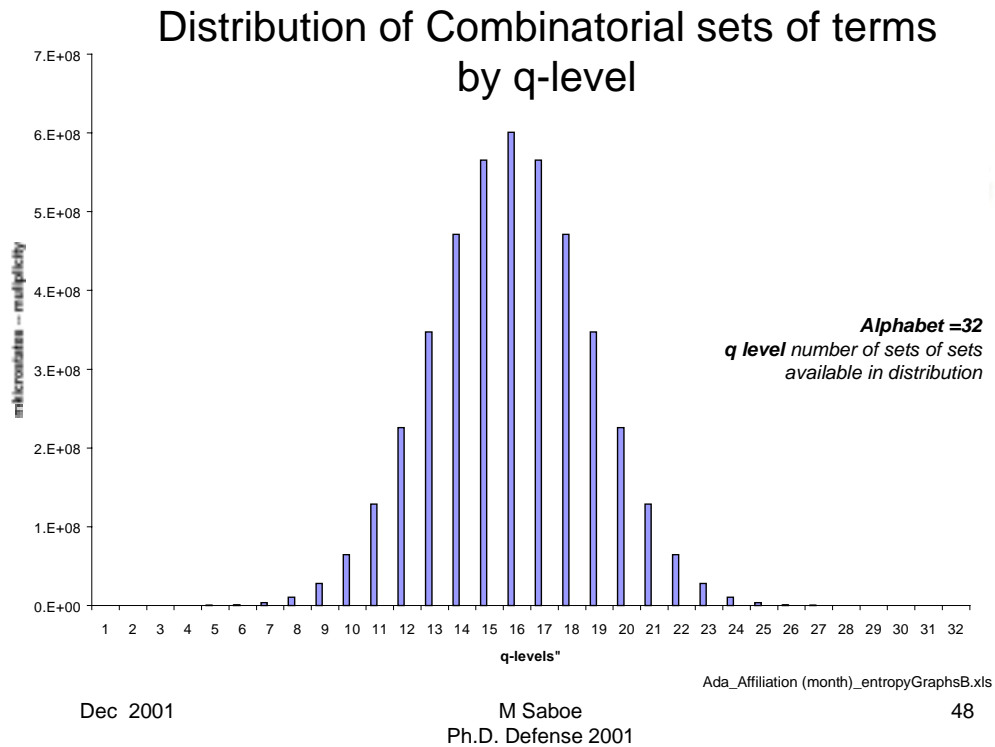
Figure III-5 Distribution of Combinatorial sets of terms

Technology sample sets would never have message sequences that are infinitely long. We are always only looking at a subset of the infinite set of sequences. They are limited by the view we take through a record identifier, an abstract, article or other work product. Ultimately, in the real world, the window message length is limited.

Technology samples have alphabets on the order of |1024| or more. The probability of pulling a set out of the sample alphabets of |4| and |32| are shown in Figure III-4 and Figure III-5. It would take a VERY, VERY large number, but not an infinite number, of messages (sets of sets) to reach maximum entropy when all of the terms are equally distributed.

Maximum entropy is a mathematical construct that defines equilibrium. It is similar to absolute zero in temperature of a physical system. It is a practical sense, it really not attainable in reasonable time scales for natural events. In a physical system, at absolute zero, we have minimum change in energy.

We expect that in our sample, relevant terms will be used increasingly. This will always skew the distribution to the left, to lower q-levels. We are never likely to get an equal distribution of terms, but in principle, it could happen.

This means that the theoretical maximum entropy is never reached in reasonable time. The maximum entropy concept is useful only as something we use to compare with. This implies we need the mechanism to determine relative entropy.

We consider each of these subsets, the primitive messages in this research. We get the count of all of the permutations for triples, and quadruples, etc. These determined composite sets of sets message data points in each technology sample. The total count of all of the terms found in a time step is used to determine the maximum entropy.

Let's now introduce the definitions for joint and conditional entropy and mutual information. These are key facets of the technology transfer models proposed.

3. Joint Entropy

Joint entropy S(X,Y) of a pair of discrete random variables *(X,Y)* with a joint distribution *(X,Y)* can be considered to be a single vector-valued random variable. The *joint probability p(X,Y)* be defined as *p(x,y)* is the probability of a joint occurrence of event *X=x* and event *Y=y*. This leads to

$$S_H(X,Y) = -\sum_{x \in \Xi}\sum_{y \in \Psi} p(x,y) \log p(x,y) \qquad (3.10)$$

which can also be expressed as

$$S_H(X,Y) = -E \log p(X,Y) \qquad (3.11)$$

4. Conditional Entropy

The *conditional entropy* of a random variable given another is defined as the *expected value* of the entropies of the conditional distributions, averaged over the conditioning random variable. If *(X,Y)~p(x,y)*, the *conditional probability* is *p(X/Y)* of outcome *X=x* given outcome *Y=y* for random variables (not necessarily independent). The conditional entropy $S_H(Y/X)$ is

$$S_H(Y|X) = \sum_{x \in \Xi} p(x) S_H(Y|X=x) \qquad (3.12)$$

$$= -E_{p(x,y)} \log p(Y|X) \qquad (3.13)$$

This is shown in the Venn diagram in Figure III-6. The mutual information is given as *I(X;Y)*.

Mutual Information and Entropy

$$I(X;Y) = S_H(X) - S_H(X|Y) \quad (1)$$
$$I(Y;X) = S_H(Y) - S_H(Y|X) \quad (2)$$
$$I(X;Y) = S_H(X) + S_H(Y) - S_H(X,Y) \quad (3)$$
$$I(X;Y) = I(Y;X) \quad (4)$$
$$I(X;X) = S_H(X) \quad (5)$$

Conditional

$S_H(X,Y)$

Joint

X (input)
$S_H(X|Y)$

$I(X;Y)$

Y (output)
$S_H(Y|X)$

$S_H(X)$

$S_H(Y)$
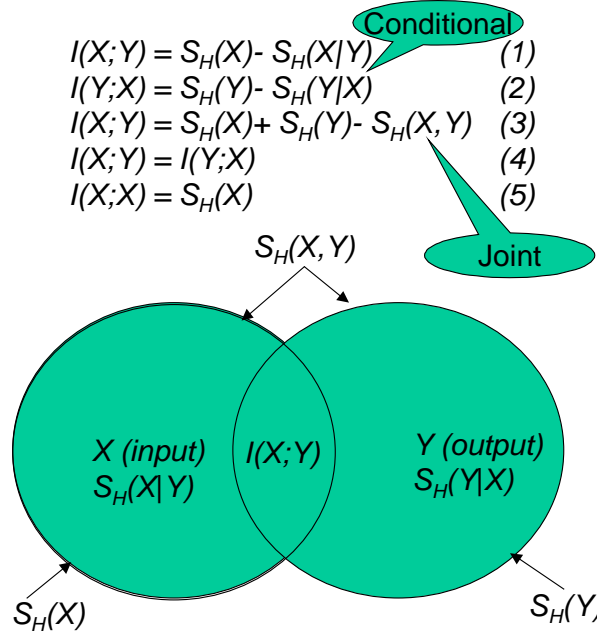
Figure III-6 Mutual Information, Joint and Conditional Entropy

Referring to Figure III-6 for the models proposed, the entropy of the vocabulary of terms at time step k is the input entropy $S_H(X)$. The joint entropy $S_H(X,Y)$ is the cumulative entropy at time step $k+1$. The $S_H(Y)$ is the incremental contribution of the time step $k+1$. The mutual information, $I(X;Y)$, can be calculated from equation (3) in Figure III-6, given the data for the input entropy, the incremental contribution, and the joint entropy. Using Figure III-6, equations (2) and (3), the conditional entropy is readily computed.

$$S_H(X) + S_H(Y) - S_H(X,Y) = S_H(Y) - S_H(Y|X) \quad (3.14)$$

Notice how S_H is dropped from the equation as we rearrange and get

$$\underbrace{S_H(X,Y)}_{\substack{S_{k+1} \\ \text{Joint}}} - \underbrace{S_H(X)}_{\substack{S_k \\ \text{Input}}} = \underbrace{S_H(Y|X)}_{\substack{\Delta S_{k+1} \\ \text{Incremental} \\ \text{new information}}} \quad (3.15)$$

The joint information is the cumulative entropy computed at time step *k+1*. This will also be the input to the next time step. The input is the pool of information (persistent messages with their constituent terms) available to the producer. On the right hand side of the equal sign, is the incremental addition of new information.

Recall from Chapter 2, that this is the feature that makes the information model, that includes a social system, different from a thermodynamic system. In a thermodynamic system with physical particles, the important feature of stochastic dynamics is the local, short-range character of the interactions. In the physical system, the number of transactions going on per unit time in a system of size *N* must be proportional to the size. That is each element can only sense its neighbors. In this system, which includes nodes of people and machines constituting a social system, this local property has to be redefined. Local is not geographically local as in a volume, but rather the volume is defined as accessible by a direct contact via a graph. Each element can simultaneously sense all of the other elements present and reachable. The studies, by Allen (Allen 1977, 1983), on influences from external sources is amplified. This leads to transition rates proportional to N^a, where the exponent *a* may be larger than unity.

5. Relative Entropy

Relative entropy or the Kullback Leibler distance between two probability masses *p(x)* and *q(x)* is defined as

$$D(p \parallel q) = \sum_{x \in \Xi} p(x) \log \frac{p(x)}{q(x)} \tag{3.16}$$

$$= E_p \log \frac{p(X)}{q(X)} \tag{3.17}$$

Similar to earlier developments, we use the convention based on continuity of arguments that $0 \log \frac{0}{0} = 0$ and $p \log \frac{p}{0} = \infty$. (Cover 1991, p18)

While it is not a true distance between distributions, it is useful to think of relative entropy as a "distance" between distributions. The mutual information which was

introduced before is the measure of the amount of information that one random variable contains about another random variable. It is the reduction in the uncertainty of one random variable due to the knowledge of the other. Assume we have two random variables X, and Y with a joint probability mass function $p(x,y)$ and marginal probability mass functions $p(x)$ and $p(y)$. The mutual information $I(X;Y)$ is the relative entropy between the joint distribution and the product distribution $p(x)p(y)$, i.e.,

$$I(X;Y) = \sum_{x \in \Xi} \sum_{y \in \Psi} p(x, y) \log \frac{p(x, y)}{p(x)p(y)} \tag{3.18}$$

$$= D(p(x, y) \| p(x)p(y)) \tag{3.19}$$

$$= E_{p(x,y)} \log \frac{p(X, Y)}{p(X)p(Y)} \tag{3.20}$$

It is important to see that the mutual information $I(X;Y)=I(Y;X)$

$$I(X;Y) = S_H(X) - S_H(X \mid Y) \tag{3.21}$$

The mutual information $I(X;Y)$ is the reduction in uncertainty of X due to knowledge of Y. By symmetry, it follows that

$$I(Y;X) = I(X;Y) = S_H(Y) - S_H(Y \mid X) \tag{3.22}$$

That is X says as much about Y as Y says about X. Since

$$S_H(X,Y) = S_H(X) + S_H(Y \mid X) \text{ we have}$$

$$I(X;Y) = S_H(X) + S_H(Y) - S_H(X,Y) \tag{3.23}$$

Also we see that

$$I(X;X) = S_H(X) - S_H(X \mid X) = S_H(X) \tag{3.24}$$

The mutual information of a random variable with itself is the entropy of the random variable.

Mutual information and the symmetry we see here is what will enable the conservation principle to be met. As *X* correlates with *Y* it is realized in the same amount of mutual information. This is easy to see in Figure III-6.

6.   Message Counting and Message content – terms

The message counting model, seen in Figure III-7, which is typically used, provides a very good correlation and is quite linear with time. This may not always be the case, but extensive studies on this data clearly showed that the linear fit was best for messages. Often, studies in the literature acknowledge that the linear fit only works after the initial slow ramp up phases. Once the initial transient is over, and the system achieves a quasi steady state, the linear fit works well.

Possibly, information theoretic and dynamical systems models can be built that enable richer analysis. The relationships to be developed should ideally be independent of the diffusion rate's function form, linear, power, polynomial, etc. While the explanation is done here for the linear model of message change over time, the general approach is developed mathematically independent of the functional form of the message rate equation. In this way, the technology under examination diffusion rate can dictate the form of the function. It turns out that linear, power or polynomial (low order) fits of the message verses time step function, all work out to be rather well behaved, and solvable in a closed form.

Traditional Model – *Message-Counting*

Traditional Method -- Count the Messages

$$y = 182.11x - 614.61$$
$$R^2 = 0.9747$$
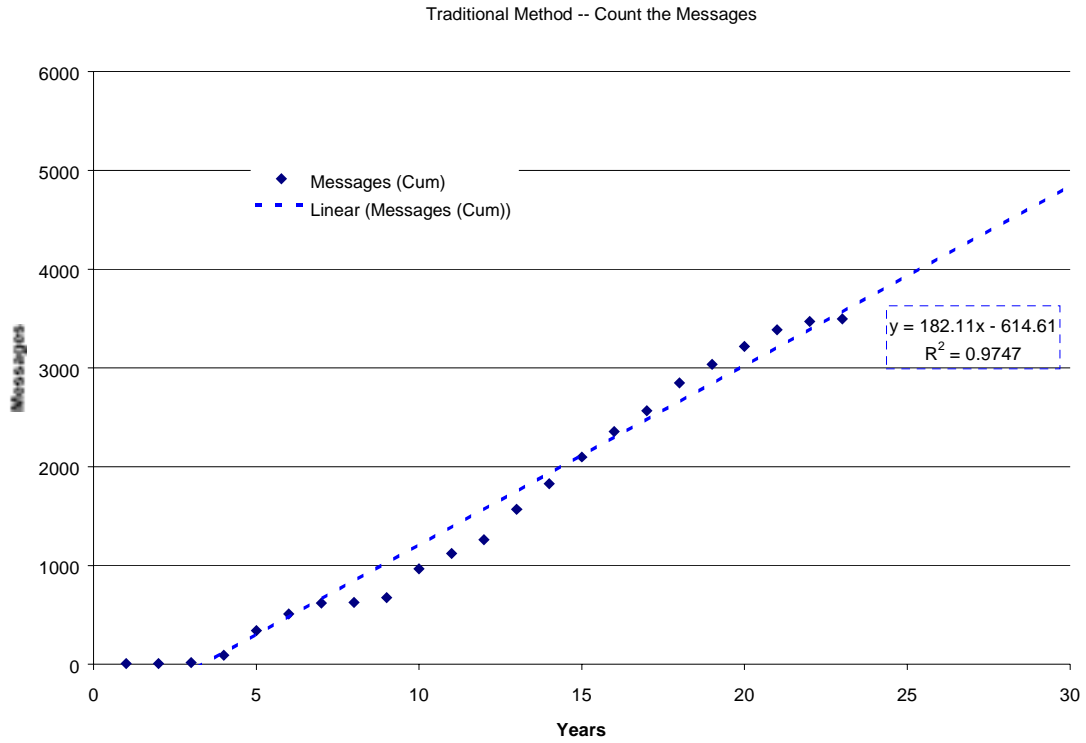
Figure III-7 Message Counting Linear model

For an information – communication model to work we need to determine the change in entropy over a time step.   In Figure III-8, we see how entropy and messages *N* vary over time.   Messages are a conserved extensive quantity, and the information entropy S_H is related to the quality of message content. The count of terms making up the messages *N* will be indicated by *n*.

Message Counting and Entropy Approach

Entropy Power fit
$y = 3.9999x^{0.1102}$
$R^2 = 0.9488$

Pretty good R^2

Message count Linear
$y = 19.366x - 381.84$
$R^2 = 0.9901$

Message Count
Entropy
Linear (Message Count)
Power (Entropy)

Messages

Entropy (bits)

Timesteps (monthly)

Jan 31 2002

M Saboe
Systems Dynamics Society
20th International Conference

24
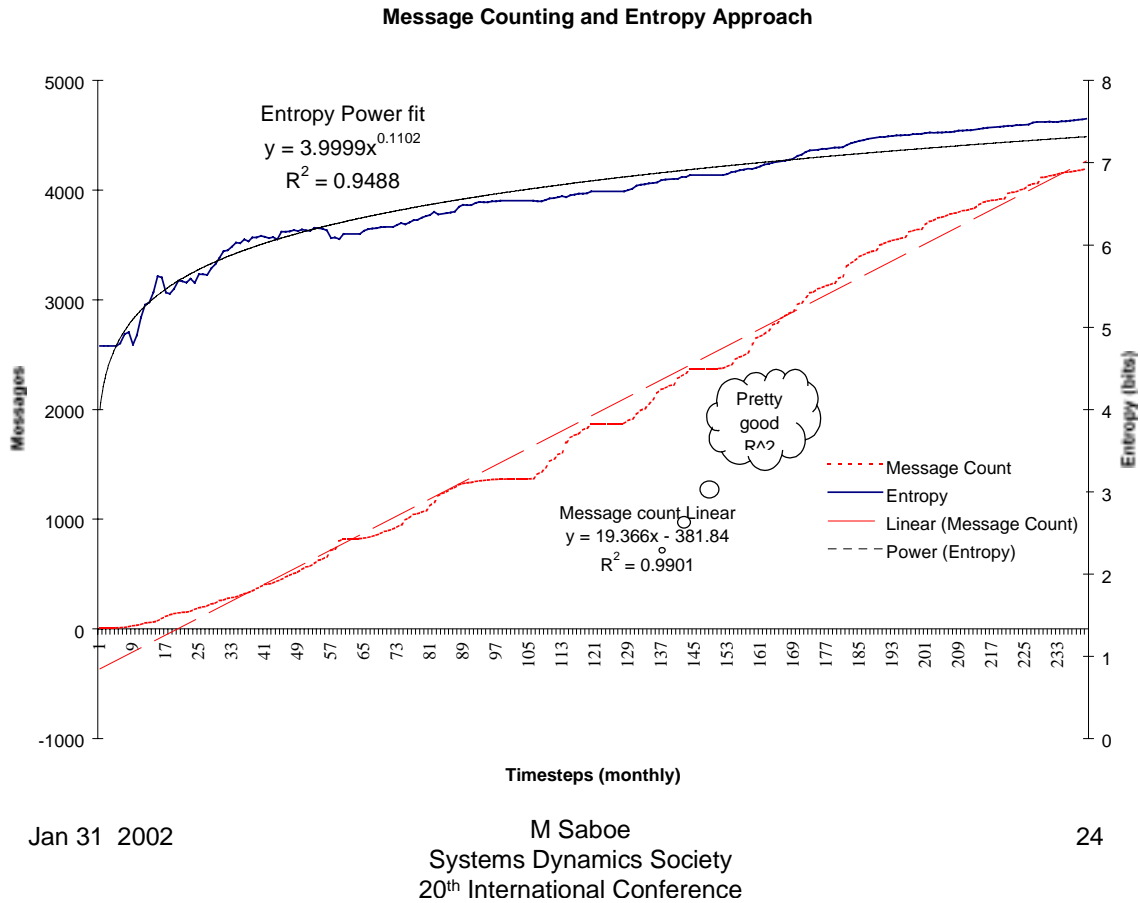
Figure III-8 Entropy and messages *N* over time

In Figure III-8, we see that we would like an illustration of the joint entropy related to technology at a given time step. Further, we would like a method to compare to different technologies, Figure III-9. This is done through the mechanism of relative entropy.

Figure III-9 illustrates two technologies. Using relative entropy, we now have a mechanism to determine how "close" these technologies are in a crude sense. But, there are other factors are work. For example, what is the mind share, the volume of nodes operating on the messages?

- 113 -

Experiment 2
Cumulative Entropy vs. Year
Java 2813 Terms, 28907 Instances, 5330 Messages, 6 Years

$y = 6.6188x^{0.159}$
$R^2 = 0.9405$

$y = 4.7404x^{0.1489}$
$R^2 = 0.9724$

▲ Java Cumulative Entropy
● Ada Cumulative Entropy
--- Power (Java Cumulative Entropy)
--- Ada 22 Years Data Project 10 Years

k **(Years)**
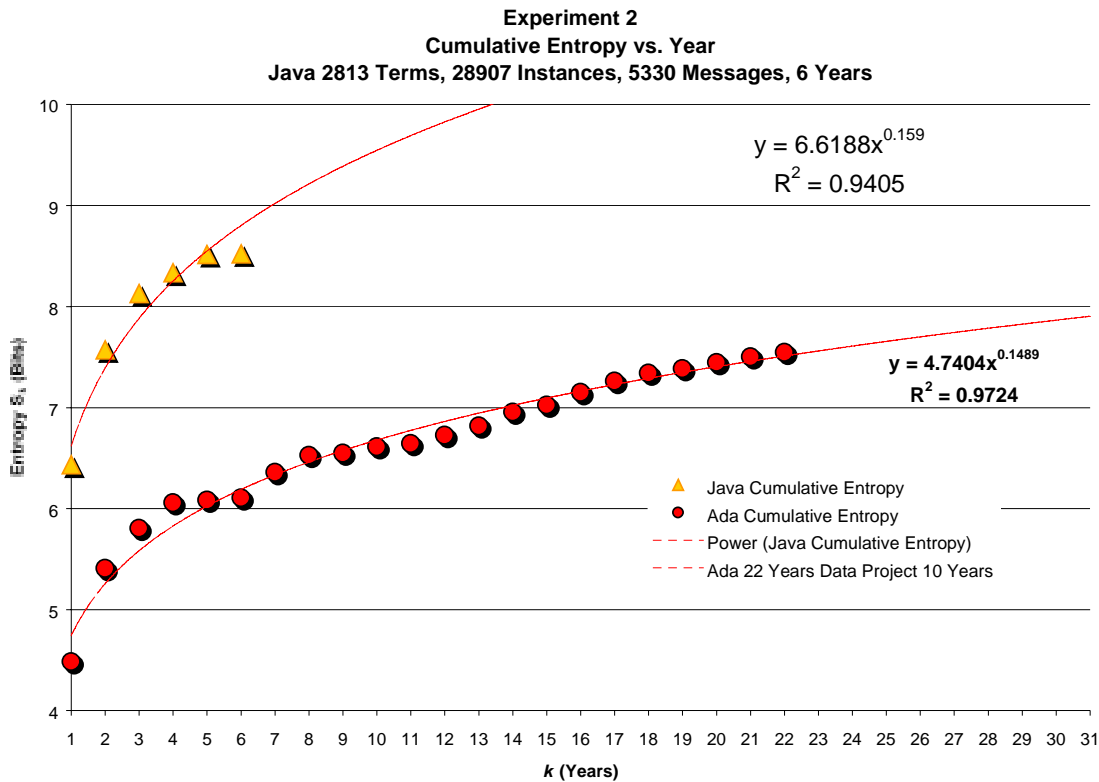
Figure III-9 Entropy vs time

7.   **Interacting Subsystems**

Let's imagine a super system (the community's world of knowledge) that consists of two subsystems. These subsystems represent what is known and what is unknown at a given time. The sum of the two subsystem's extensive variables messages *N*, and nodes *V* is constant. Here the conserved extensive variable properties are *N* messages, and the sum of all the nodes, *v,* which is the volume *V*. This will define a control volume. The rate of change follows the rate we would expect if this were modeled as an open system during these time steps. Now we will take a virtual partition and have it progress expanding subsystem A to the right. As this partition passes over some nodes, effort is made by the nodes and they "discover" a term. Terms *n* are the internal pieces of the messages *N.* Terms are defined as primitive messages. Counting terms is similar to counting the messages, but at a finer granularity. The nodes stimulate and change the

internal configuration of the system by converting an undiscovered term (a null) into a communicated discovered term.

This can be seen in Figure III-10. On the left hand side we see "!!!s" representing terms that have been discovered (answers), on the right hand side of the partition we see "???s" representing terms that are yet undiscovered (questions). The Venn diagrams indicate the subsystems A and B, joint, conditional entropies and mutual information, as illustrated earlier. Examine what this looks like with a sample alphabet as in Figure III-11.

The nulls {} are terms that have not yet been discovered at the frontier of the research in time. We might ask, if the null or "???" terms really exist and are representative of the real world. Researchers or any node that builds a work product or messages is actually working toward a yet unrealized collection of answers "!!!". They envision the potential combination of terms (primitive sets) that can make a representation of the goal directed, objective work product that is desired. Certainly, during the period of time when a node is developing the answer, the term under question exists. Desires, although they are not representational states, do have an object, something they are a desire *for*. This is the "???" term. Desires[13], like beliefs, are intentional states (Drestske 1988 p130). The nulls represent the "???" questions desired by research.

In this simplified example, we are assuming a fixed set of terms in the alphabet, and a fixed number of nodes. This will permit the development of the general relationships between extensive and intensive variables in a state equation. Later, once we have seen these relationships, we can start with an initial condition representing the number of terms and nodes known up to that time. Then we add more vocabulary to the system or more author nodes any way we wish. The rate of change, when reduced to per node, and per term (specific) extensive variable rate will be expected to remain for the

---

[13] Not all desires are realizable. Some desires inherit the referential opacity from the beliefs and other desires from which they are derived. "Desires, are like beliefs, referentially opaque. The belief that s is *F* is not the same as the belief that *t* is *G*, although *s=t* and although the predicate expressions *"F"* and *"G"*, are true of, or refer to exactly the same things." (Drestke 1988 p130). The same is true of an object desired. In the ancient Greek play of Sophocles, Oedipus wants to marry Iocasta, but does not want to marry his mother (and perhaps even wants *not* to marry his mother), despite the fact that Iocasta is his mother.

future (open system) similar to the rates for the historic (closed) subsystems.   This permits the design of a desired solution in the form of an engine.

Interacting Systems
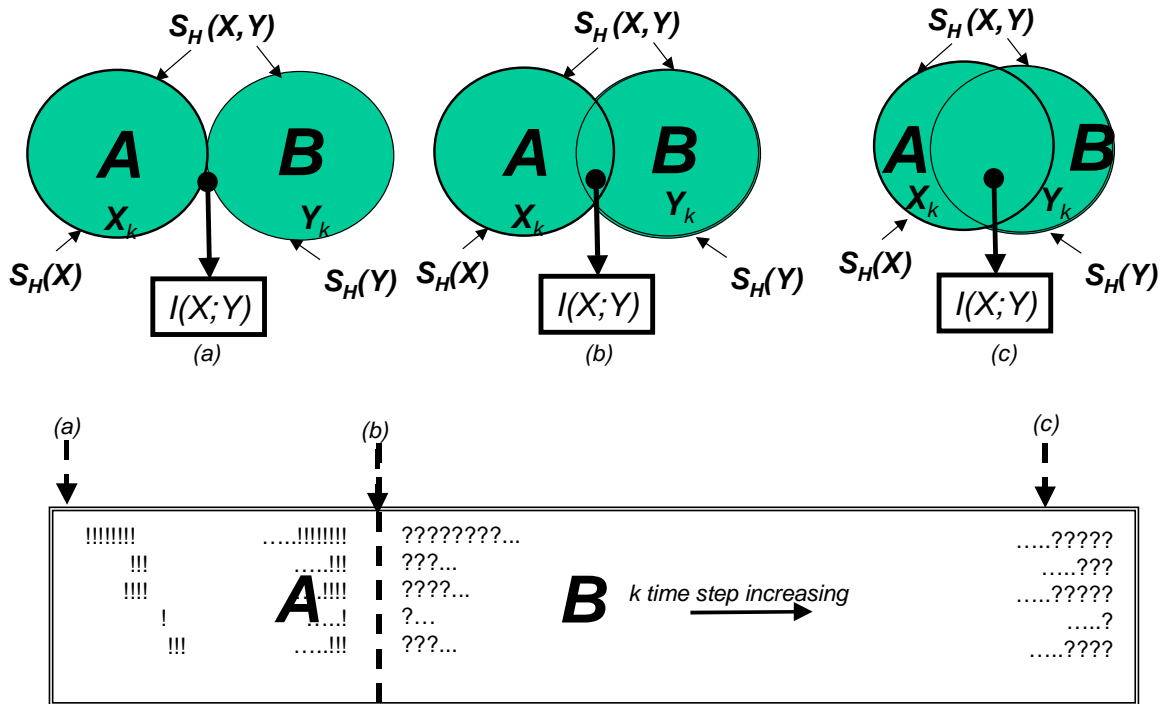
Figure III-10 Interacting Systems A and B

In Figure III-12[14], we see that as system A expands, the number of terms discovered increases, at the same rate that the number of terms undiscovered decreases. This model satisfies our conservation principle for extensive quantities.

Next, in Figure III-13, we examine the entropy relationship. The horizontal line at the top of the figure is the joint entropy of the system. Since this is a closed system, this is not changing, however, the internal distribution will change. That entropy related to subsystem A will increase as the are more and more choices to make in order to get complete information.   Subsystem B will decrease from a high entropy (all of the unknown terms) to a lower entropy as there becomes less and less left to be discovered.

---

[14] The charts in this section represent initial data to illustrate the general relationships.   Actual

The lower curve shows the mutual information. When the distance between the center of the two probability masses, or subsystems, decreases, there is a higher correlation.

FirstOfPubYear	CountOfdescr	descriptor_text	1979	1980	1981	1982	1983	1984	1985	1986	1987	1988	1989	1990	1991	1992
1980	3	procedure-oriented-languages	0	3	3	3	3	3	3	3	3	3	3	3	3	3
1981	140	Ada	0	0	14	16	22	30	38	40	51	57	101	101	109	140
1983	4	systems-analysis	0	0	0	0	1	1	1	1	1	1	1	1	1	4
1981	39	software-engineering	0	0	1	2	3	11	11	18	19	21	36	36	36	39
1981	32	program-compilers	0	0	5	6	9	12	12	14	18	19	30	30	31	32
1981	8	operating-systems-computers	0	0	1	1	1	3	3	3	4	5	7	7	7	8
1984	18	distributed-processing	0	0	0	0	0	1	1	1	2	4	13	13	15	18
1986	34	software-tools	0	0	0	0	0	0	4	5	8	22	22	23	34
1984	27	real-time-systems	0	0	0	0	0	1	1	1	2	4	8	8	12	27
1989	8	object-oriented-programming	0	0	0	0	0	0	0	0	0	0	3	3	3	8
1988	11	scheduling	0	0	0	0	0	0	0	0	1	5	5	6	11
1981	23	programming	0	0	1	1	2	6	6	7	9	9	17	17	17	23
1991	9	software-metrics	0	0	0	0	0	0	0	0	0	0	0	0	1	9
1992	4	standards	0	0	0	0	0	0	0	0	0	0	0	0	0	4
1986	3	structured-programming	0	0	0	0	0	0	0	1	1	1	2	2	2	3
1989	6	fault-tolerant-computing	0	0	0	0	0	0	0	0	0	4	4	5	6
1989	2	software-reliability	0	0	0	0	0	0	0	0	0	1	1	1	2
			///	///	///	///	///	??	??	??	??	??	??	??	??	??

Figure III-11 Subset of an alphabet in two interacting systems !!! and ???

Figure III-12 Messages in two subsystems

---

equations for a specific technology are shown in Chapter IV, and the appendix.

Entropy vs Messages
Two Subsystems

Entropy 2 Interacting Systems A and B

$S(X)\_A = -5E\text{-}05n^2 + 0.0228n + 4.8832$

$R^2 = 0.925$

$S(Y)\_B = -0.0002n^2 - 0.0021n + 7.6107$

$R^2 = 0.9925$

$S(X|Y) = -1E\text{-}04n^2 + 0.0312n + 4.6377$

$R^2 = 0.7959$

S Entropy

- - - - S(Y)_B
——— S(X)_A
——— S(X,Y) Joint A and B
——— S(X|Y) !!! Less Mutual Info
- - - Poly. (S(X|Y) !!! Less Mutual Info)
- - - Poly. (S(X)_A)
- - - Poly. (S(Y)_B)

Two subsystems
are at an equilibrium point
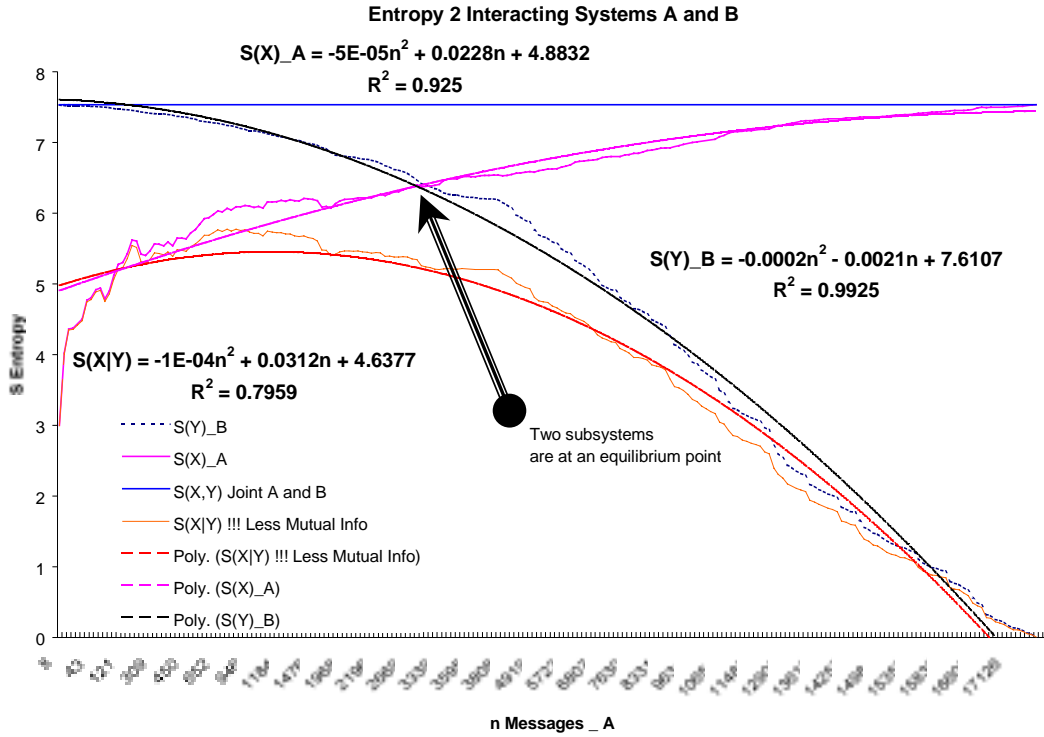
n Messages _ A

Figure III-13 Entropy vs. Messages Two interacting Systems

Following reasoning similar to that used in statistical, and condensed particle physics (Schroeder 2000) (Fraundorff 2000), we can find some useful relationships. The slope of the curves of the two subsystems gives us some important information about thermal equilibrium. Recall from the canonical ensemble discussion of free energy, that the temperature T is the parameter controlling free energy, or the conserved property. In this case of messages, we can write

$$\frac{1}{T} = \frac{\Delta S_H}{\Delta n} \qquad (3.25)$$

So the temperature is related to slope of the change in entropy verses change in messages curves. When the curves in the figure cross over, the system is at an

equilibrium point. Let's look at a general relationship that shows the increase in one system is related to the negative slope or, the decrease in the other.

$$\frac{\Delta S_A}{\Delta n_A} = -\frac{\Delta S_B}{\Delta n_A} \qquad (3.26)$$

The incremental change in S_A, divided by the change in n_A messages, is equal to the change in entropy, S_B, for system B again compared to the change in the conserved quantity, in this case n_A. Rewriting we get

$$\frac{\Delta S_A}{\Delta n_A} + \frac{\Delta S_B}{\Delta n_A} = 0 \qquad (3.27)$$

The second term has a B in the numerator and A in the denominator. Δn_A is the same as $-\Delta n_B$, since what we discover in messages is the same as what is removed from the undiscovered system. We can rewrite this for a system at equilibrium as

$$\frac{\Delta S_A}{\Delta n_A} = \frac{\Delta S_B}{\Delta n_B} \qquad (3.28)$$

The thing that is the same for both systems when they are at thermal equilibrium is the slope of the entropy message graph. This slope must somehow be related to the temperature of the system. The 2nd law of thermodynamics tells us that the conserved property will tend to flow *into* the subsystem with the *steeper* entropy vs. message graph, and *out* of the object with the *shallower* entropy vs. message graph (Schroeder 2000 p87).

According to Schroeder, the former "wants to" gain the free conserved property (messages) in order to increase its entropy. If there is an imbalance between the two subsystems, the latter doesn't so much "mind" losing a few messages (since the entropy will not decrease much. A steep slope must correspond to a *low* temperature, while a shallow slope corresponds to a *high* temperature.

Now we can see in the lower curve of Figure III-14, the relationship of the temperature (the right hand *y*-axis) of sub-system A as the partition moves over the time steps. More activity increases the temperature. The temperature is measured in degrees as we would in a physical system; however, these degrees are developed from

information units. This is "the" fundamental temperature unit developed from the relationship of entropy, and the conserved quantity.

Note that there are temperature fluctuations. This is consistent with Prigogine's observation about evolving systems. A dynamical system will help explain these fluctuations.

Pressure and Temperature
vs timestep
Temperature and Pressure vs Timestep

$$P(A)= 0.0058k + 1.1055$$
$$R^2 = 0.9642$$

$$T\_A(k) = 152.92k - 4183.7$$

Legend:
- Messsages per Node (A)
- Temp (A) from Polynomial
- Linear (Temp (A) from Polynomial)
- Linear (Messsages per Node (A))

Y-axis (left): Pressure (A) Messsages per node
Y-axis (right): Temperature (B) (Saboe Degrees)
X-axis: *k* Timestep (months)

M Saboe 1/25/02
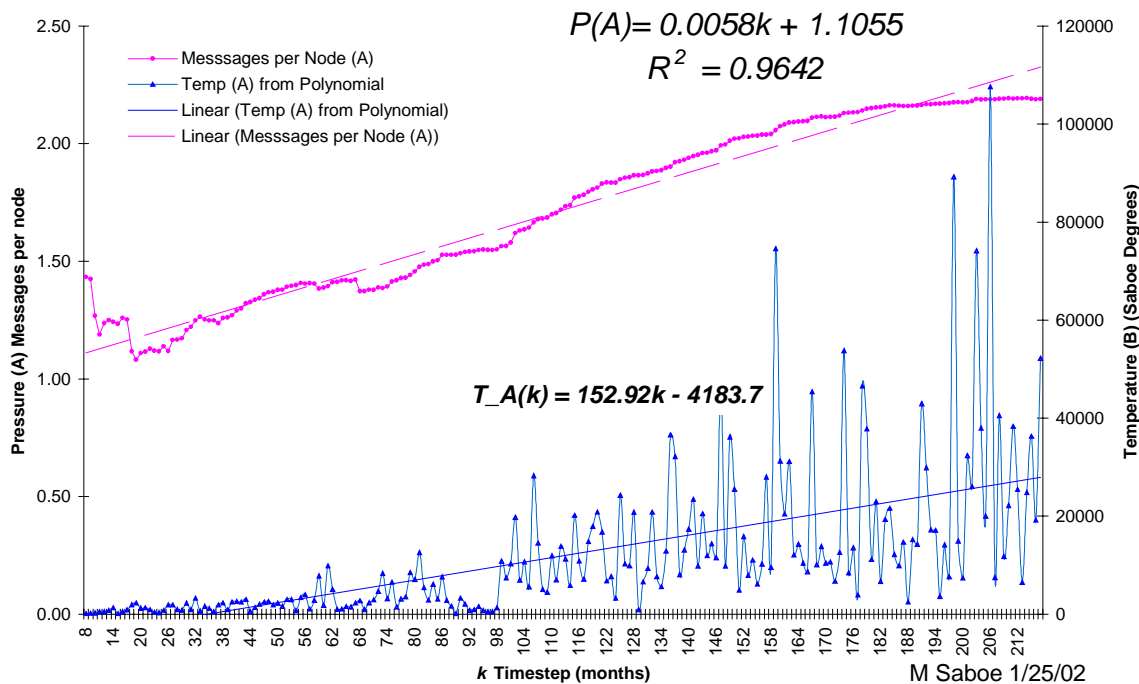
Figure III-14 Pressure and Temperature ^{o}Saboe© vs. time – two interacting systems

Pressure is defined as the <messages> processed per node, where the <messages> represent the average in the time step per node. The important observation is not necessarily the form of the equations or the goodness of fit, rather, that *the pressure can be seen to increase as the temperature increases*. While messages are not physical molecules as in a thermodynamic system, they seem to behave as a gas might, as the temperature goes up the pressure goes up.

Figure III-15 shows the relationship directly between pressure and temperature. This was developed by taking the curves from Figure III-14 and setting them both equals to k. Then the Pressure $P(T)$ as a function of temperature is determined.

$$P(k) = m_P + b_P \qquad (3.29)$$

$$\frac{P(k) - b_P}{m_P} = k \qquad (3.30)$$

Similarly, solve for k as a function of T.

$$T(k) = m_T k + b_T \qquad (3.31)$$

$$\frac{T(k) - b_T}{m_T} = k \qquad (3.32)$$

Then we get

$$\frac{P(k) - b_P}{m_P} = \frac{T(k) - b_T}{m_T} = k \qquad (3.33)$$

$$P(k) = \frac{m_P}{m_T}(T(k) - b_T) + b_P \qquad (3.34)$$

When plotted in Figure III-15 is the tight set of points indicating as temperature increases, pressure increases. Figure III-15 shows the raw data points as well. These fluctuate around the $P(T)$ calculated data, would be expected.

Pressure Temperature

Pressure vs Temperature Saboe Degrees
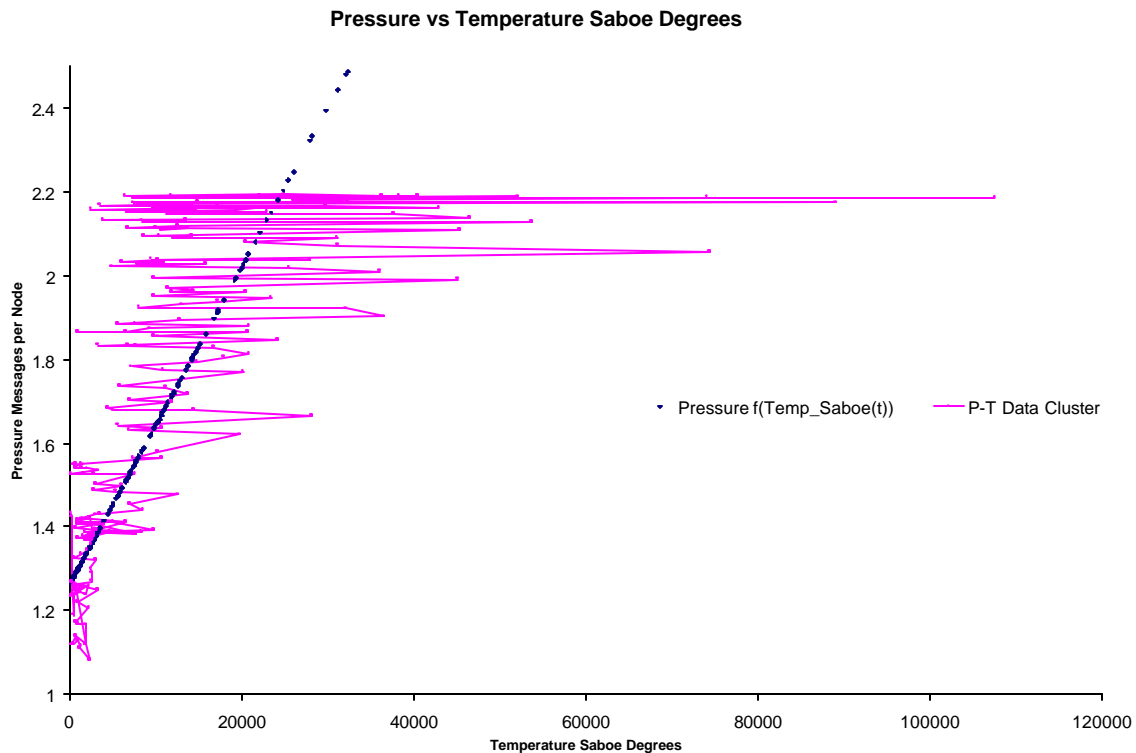
Figure III-15 Pressure vs. Temperature ^{o}Saboe ©

The application that was written to solve this relationship was also developed for the cases of power, and 2nd order polynomial. In the application code written for this project all of the permutations, linear pressure as a function of time, and power temperature as a function of time, power pressure vs. time, polynomial temperature, etc. were developed. Future efforts will automatically pick the best fit for the technology under examination and develop the *P(T)* function from that.

Typically, a state diagram viewed by engineers is a temperature – entropy, or *T-S* diagrams, (recall Figure II-13). The lower curve of Figure III-16, the *T-S* is illustrated. This is the entropy of sub-system *A* with entropy (upper *x* axis) and temperature (secondary *y* axis on the right). Since this system was not engineered, we do not expect to see anything approaching isentropic expansion, or a constant pressure, temperature increase.
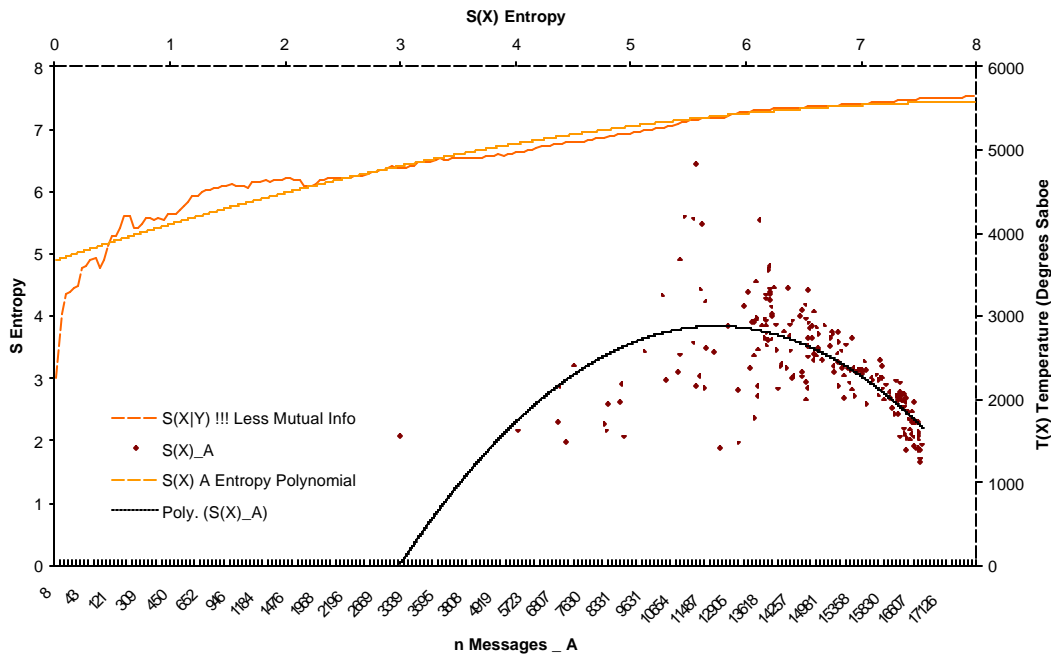
Figure III-16 Entropy -- Messages, and Temperature – Entropy

The figure also shows entropy of subsystem *A* (left *Y* axis) and messages *n* on the *x* axis. From this information in a closed system, we can see the trends for a given technology over time. In a way, we have the ability to define the heat capacity[15] (say C_p, heat capacity at constant pressure, C_v, heat capacity at constant volume, or the ratio of the heat capacities, $g = \dfrac{C_P}{C_V}$) in bits. This allows us to move to an open system, like an engine, add nodes, *volume*, and increase message flow. We can then compute our effort required from a desired "engine" to develop a technology to arrive at a given time.

$$\Delta U = \dot{n} C_p \Delta T \qquad\qquad (3.35)$$

---

[15] Heat capacity for sate equations are property relations and as such are independent of the type of process. C_p is the amount of "stimuli" transferred to a system per unit "message" per unit degree rise during a constant pressure process.

This says the change in the "internal" system energy U is related to the message flow rate $\dot{n}$ (messages per time step), the heat capacity and the change in temperature from a high temperature to a low temperature.

This also implies the equivalent of Carnot's cycle, which can tell us the maximum efficiency we can expect.

Since "internal" system energy U is introduced, let's look at this a bit further. This is related to the internal structure distribution of the terms. The set of sets of terms, reduced to primitive message combinations follows a Boltzmann distribution, Figure III-17. On the x axis, is the q-level, representing the number of terms in set. The lower curve on the y-axis is the frequency of sets. The upper curve assigns a weight to each set. The weight simply changes the quantity by a constant. We can ignore it for the purposes of these analyses. It is interesting to note, as well, that these curves plotted over the time steps examined (up to 21 years) essentially remain stationary (Figure III-18).

This change in q-levels (microstates) can be addressed by equations (2.9) and (2.10). This permits conjecture in the deeper meanings of the distribution of terms. Further, state transitions moving from one q-level to another, must somehow be affected by an impulsive stimuli of some sort. That implies both the notion of kinetic and potential "energy". This is the result of stimuli of researchers expending effort to combine primitive terms and or sets, composing more sets of sets. "Discovering" new single terms, the first time a ??? augments the vocabulary also is the result of a change of state from a {}, null, to the first instance of an answer. !!!. This too takes effort. These topics are subject for future research.

q level Distribution
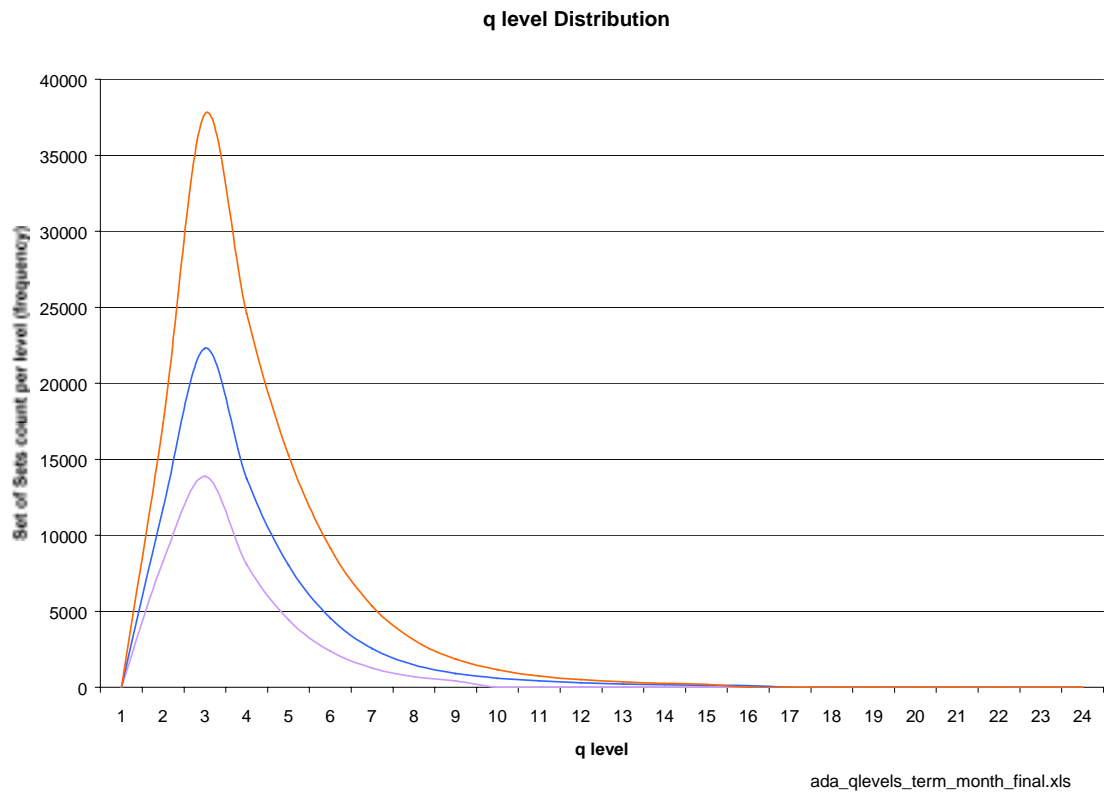
Figure III-17 Boltzmann Distribution of Sets of Terms (primitive messages)

Ada Distribution of Messages by q-level

q level Distribution
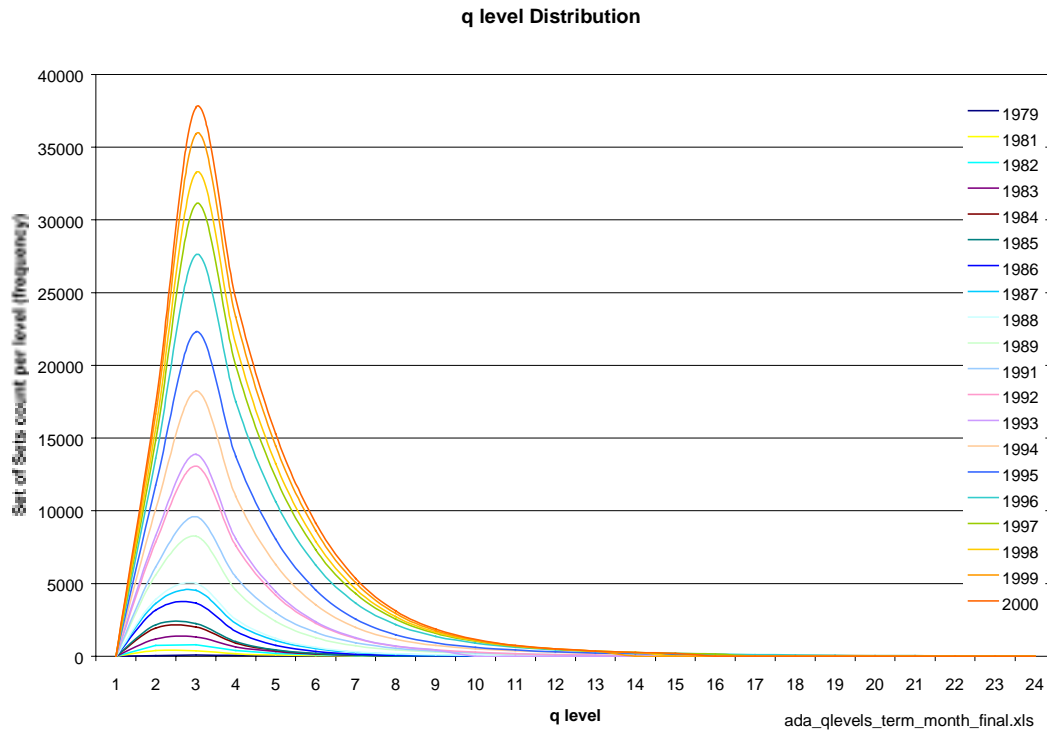
Figure III-18 Set of sets distribution over time steps by q-level

8.   Technology Transfer Channel Elements

We consider two cases. The deterministic case represents the microscopic level of the model in the system, and the stochastic case represents the macroscopic system view. So far, we have only addressed the macroscopic case. The deterministic case would occur at the micro level in a program, or a system made up of nodes consisting of a family of machines. A stochastic system consists of a population, coarsely partitioned at the macroscopic level. This is a system made up of a social environment consisting of people and organizations. The *TechTx* models address the more general case of the stochastic system of nodes consisting of people, organizations and machines.

We define the community, the macro structure, as a set of performers that produce output. An organizational is made up of a set of the performers with which they are affiliated. We can think of the micro level in terms of the performers. The organizational level is in between the macro and micro levels and can be thought of as an ensemble of affiliated performers. We can observe individual output from the data. Each record contains primitive messages published by a performer, x_{i_1} contributes information to the community. This is defined as follows.

$$\overline{X} = \bigcup_{i=1}^{p} \overline{X}_i \text{ is the } community \tag{3.36}$$

$$\overline{X}_i = \left\langle x_{i_1}, x_{i_2} \ldots x_{i_n} \right\rangle \text{ where } x_{i_1}, x_{i_2} \ldots x_{i_n} \text{ are the } performers \text{ of the } i^{th} \text{ } organization$$

$$\tag{3.37}$$

$$\overline{X}_i \text{ is the } i^{th} \text{ organization, and } i = 1.. p \tag{3.38}$$

The output entropy is allocated from the message to individual author subset performers from the empirical data. This micro level is then summed up and allocated to the to the affiliated organizational level. The organizations are banded based on a distribution of the cumulative number of published messages.

We consider a family of nodes (machines, and people – the atomic level), making up organizations (the molecular level), and a community (macro level). In a band, we assume all of the nodes have the equivalent properties, i.e. each organizational node, comprised of performing author nodes, are statistically equivalent. Figure III-19 and Figure III-20 illustrates a node taking information in as input $S(X)$, performing some transformation, $F(X_k)$, to produce more messages (work products). Part of the output is expanding the mutual information $I(X;Y)$ intersection of the Venn diagram, and part is augmenting the vocabulary. This augmentation is the conditional probability $S(Y/X)$, as we saw from equation (3.15).

Node Input and Output
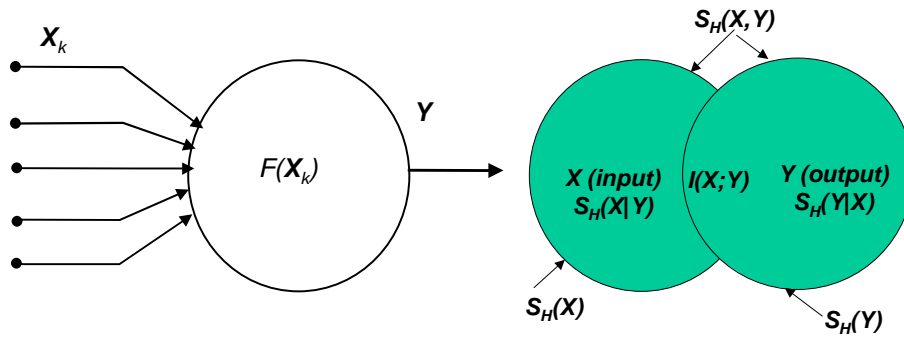
Figure III-19 Input being converted via a transfer function to output

Node Transform of Input to Output
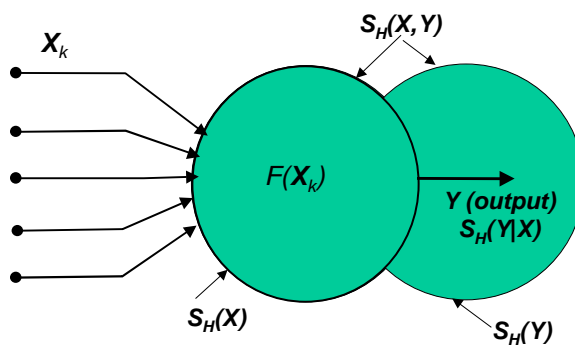
Figure III-20 Node transform of Input to Output.

The initial band determination is computed based on the accumulation of experience of executing tasks, i.e. publishing messages. The most prolific performers are banded together based on the average number of messages produced over the period examined. Later the learning, or performance index is computed for each band at every time step from the beginning of the data set to the (current) performance time step. An example of the distribution is shown in Figure III-21.

We will perform a coarse partitioning of the performing organizations into four bands. Further, partitions are possible, however this is sufficient to demonstrate the approach. The "A" band consists of all of the organizations that were beyond 3σ in the rate of production of messages in the sample for a given technology. The "B" band are the organizations in the 3σ partition. The "C" band contains the organizations with a message production history in the 2σ partition, and the "D" band are all of the organization below 2σ in performance.

Productivity Distribution (Sample)
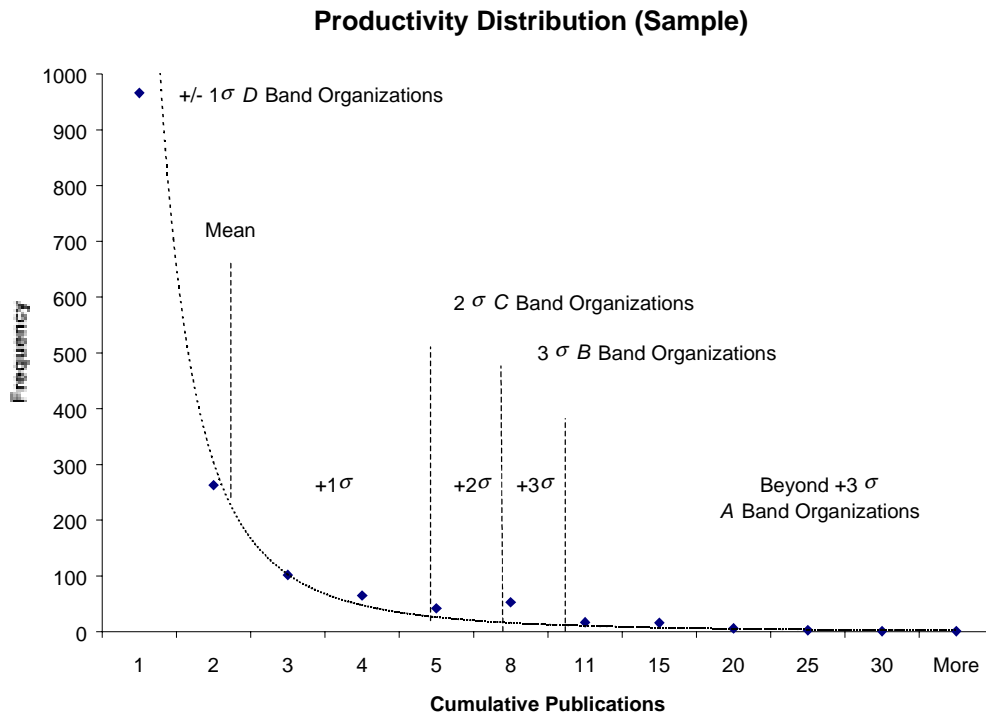
Figure III-21 Organization Distribution into Cumulative Task Performed Bands

Our problem is to realize, or at least to approximate, a given system, which we call the true system, by a model. We adjust the parameters values based on a number of examples provided by observation of the true system.

The analyses of the partitions can proceed exactly as the analysis of the macro level community. This is the beauty of the partitioning. We only have to be cautious of combining bands when the counts of terms, (multiplicity of states) are "local" to the band under examination. We count messages in a band and develop the probabilities, and hence the entropy of the band is based on the total number of messages in the band. In order to aggregate bands, we consider this entropy the band's contribution to the total (all bands) entropy. There is an entropy contribution simply resulting from the partition. This contribution varies every time step based on the internal organization of the messages, constituent terms, and nodes.

Node Input and Partitioned Output

$S_H(X,Y)$

X_k

$F(X_k)$

Y_A (output)
$S_H(Y|X)$

I_A

X (input)
$S_H(X|Y)$

I_B

Y_B

Y (output)
$S_H(Y|X)$

I_C

Y_C

I_D

Y_D

$S_H(X)$

$S_H(Y)$

$|\Omega|$ *multiplicity of terms in Technology World*

$|\Omega_A|$ *multiplicity of terms in A Band*

...

$|\Omega_D|$ *multiplicity of terms in D Band*

C_i *is contribution of Band i*

$$C_i = \frac{|\Omega_i|}{|\Omega|} S_{H_i} + \frac{|\Omega_i|}{|\Omega|} \log \frac{|\Omega|}{|\Omega_i|}$$

$S_{H_i} = $ The entropy calculated locally for Band i
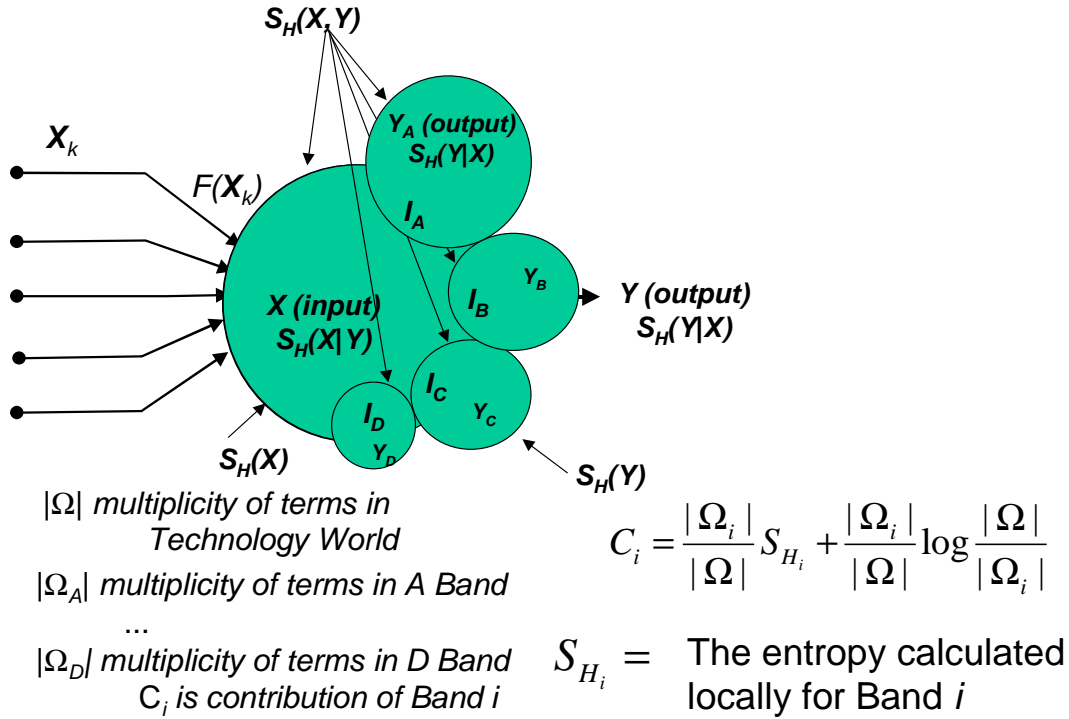
Figure III-22 Partitions of output into bands. Contribution to the Community

Each band, i, provides a contribution, C_i, to the community entropy. The local band entropy S_{H_i}, must be scaled based on the multiplicity Ω_i of terms in the band to the multiplicity Ω of terms in the world. The community, which is sometimes referred to as the technology's "world" entropy is the sum of the contributions.

$$S_{H_{world}} = \sum_{i=1}^{n\_bands} C_i \tag{3.39}$$

where $C_i = \dfrac{|\Omega_i|}{|\Omega|} S_{H_i} + \dfrac{|\Omega_i|}{|\Omega|} \log \dfrac{|\Omega|}{|\Omega_i|}$ (3.40)

This relationship permits aggregation of previous results on a subset of a community with more information later without having to rerun the entire world and all previously analyzed bands. All that is required is the count of the instances of terms in a

band and the count of the number of instances of terms in the world, augmented by these terms.

Later extensions to be considered would address all of the various combinations of author nodes producing a message. For example, the x_{i_n} performers could be represented as combinations of authors producing a record (which as was pointed out, is broken down into its primitive messages at various q-levels). Additionally, we could assume that if there are three authors on a record, they represent 2^3 possible author subsets – nodes. Each subset is a legitimate combination producing the messages. This distribution develops in exactly the same way as the term distribution of sets of sets as developed. The ability to calculate the contribution with a ratio of the local system instances to the microstates of a larger or smaller system, it was often useful to count instances of states. By computing the entropy locally, these chunks can be combined with other subsystems often with out additional computation.

C. COMMUNICATION AND CONTROL MODEL

What has been described thusfar is an information-theoretic view of the macro world, and a method to partition the world into bands. For now we will continue to work at the world level, however, recognize that we can partition the world and demonstrate the same relationships. Now we marry up a dynamical systems model with the information-theoretic model. When both models stabilize, at a rate represented by equations of the same *form*, we have moved in the direction of a match between the macro (continuous model) and micro (discrete) model. The true system may be considered modeled when we *tune* parameters in the discrete model and align the entropy and conserved property evolution as a function of time.

1. State Space Representation

We can represent a map of state space of a dynamical system. Maps represent a simplified form of dynamics that makes it easy for us to compare the individual level of description (the trajectories) with the statistical description. Contrary to what occurs in ordinary dynamics, time in maps acts only at discrete intervals. Recall that the bakers' transformation[16] example illustrates the mixing of a spot of sauce on a piece of dough, then folding and stretching of dough. In technology maturation, a node is locally taking in a chunk of dough, messages out of the pool of messages persistent in history, and mixing them along with new information, e.g. a new term, which represents yet another spot on the dough. These areas contain remnants from bakers' transformations of other nodes that performed the mixing and adding function throughout time. A performing node may perform a number of iterations. Other nodes also perform the folding, stretching and mixing function. The mixing may occur before and concurrent with mixing at a node. The nodes successively repeat the iteration action. We represent this with dynamical system maps, with discrete time *n*. Let X_{n+1} be the function that represents the value corresponding to the application of *n* bakers' transformations.

---

[16] Details are provided in Prigogine 1989 p200-204, a summary is shown in the appendix, p288.

$$X_{n+1} = F(X_n) \qquad\qquad\qquad (3.41)$$

The various functions X_n are functions of internal time. The internal time is an *operator*[17] like the one used in quantum mechanics (Prigogine 1989 p198). The age of partition X_n is the number n of iterations i that are to be performed to go from X_o to X_n.

For ordinary differential equations, (continuous in t) this is

$$\frac{dX(t)}{dt} = G(X(t)) \qquad\qquad\qquad (3.42)$$

In both cases, X is a vector[18]. The term orbit will frequently arise in the following discussions. The orbit of a dynamical system is that sequence of points in the state-space phase plane that corresponds to successive time steps in the system. An orbit is generated for a map and $X(t)$ for the differential equations when given an initial value of X (at $n=0$ for the map, and $X(t)$ for the differential equations).

Figure III-23 shows a map of the state space. The legend shows the Java entropy map marked with a triangle (▲) and a dashed line as the upper set of points. The marker represents data, the dashed line is an indicator of the curve that would fit the data. In this case, it is in the general form of a power function where $y=3.46x^{44}$ with an $R^2=.9934$. Where $S_{H_{k+1}} = bS_{H_k}{}^m$ is the specific equation.

Similarly the circle (◯) and dashed line legend are for the Ada points, the lower set of points. In this case, the state space map is shows that the data is oscillating in the early stages. This shows that the vocabulary and threads of research have not settled down at first. Based on observation, see Figure III-23, as the entropy increases, but at declining rate, the data starts to approach the $y=x$ line. The spacing between each data point gets closer together. This indicated that the data is moving toward a stabilizing attractor basin.

---

[17] Operators, eigenfunctions, and eigenvalues are briefly summarized in the Appendix p238.

[18] We use the form $X_{n+1} = F(X_n)$, where X is a p-dimensional vector.
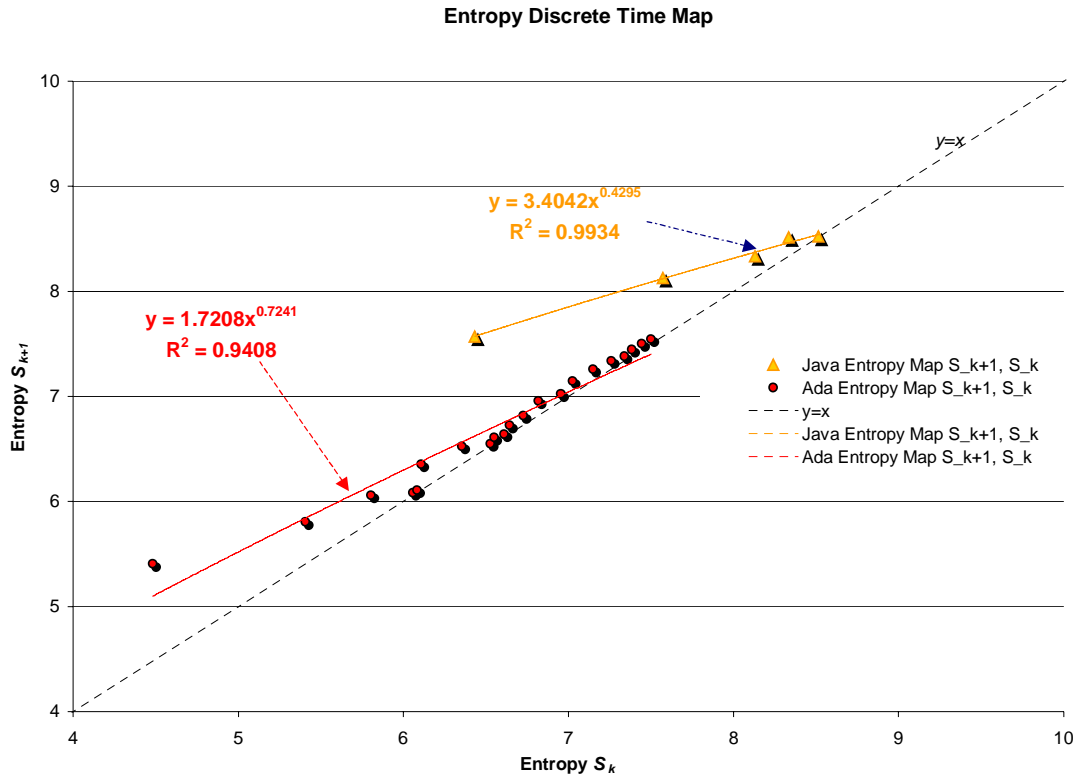
Figure III-23 Java and Ada State Space Finite Difference Map S_{k+1}, S_k

The discussion here looks at the attractor of these dynamical systems, since we are making the conjecture that the model for technology transfer, or evolutionary development can be represented in this form. If the system being evaluated attracts, then the evolution is going toward stability. We'd like to be able to say something about the confidence as the system stabilizes after initial conditions die out.

The attractor is something that attracts initial conditions after the start up transients fade. An *attractor* is a compact set, *A*, with the property of *A* such that for almost every (see Farmer 1983) initial condition the limit set of the orbit as *k* or *t* $\rightarrow +\infty$ is *A*. So almost every trajectory in the neighborhood of *A* passes arbitrarily close to every point in *A*. The basin of the attraction of *A* is the closure of the set of initial conditions that approach *A*

The eigenvalue of the characteristic equation has a relationship to entropy. This relationship is through the Lyapunov exponent, which gives the stretching rate per iteration averaged over the trajectory. Using the bakers transformation a completely deterministic dynamical system can yield results that appear completely random. The bakers transformation also has the property of all dynamical systems, recurrence. The bakers transformation is *invertible, time reversible, deterministic, recurrent* and *chaotic.*

Bakers Transformation

(a)   *(b)*   *(c)*

$$\begin{bmatrix} x_{k+1} \\ y_{k+1} \end{bmatrix} = \begin{bmatrix} 2x_k \\ y_k/2 \end{bmatrix}, \quad 0 \le x_k < \frac{1}{2}$$

$$= \begin{bmatrix} 2x_k - 1 \\ y_k/2 + \frac{1}{2} \end{bmatrix}, \quad \frac{1}{2} \le x_k \le 1$$

- Repeated doublings in the *x* direction and halving in the *y* direction leads to rapid mixing.

- The mapping is completely reversible. Run backwards, the doubling occurs in the *y* direction and halving occurs in the *x* direction
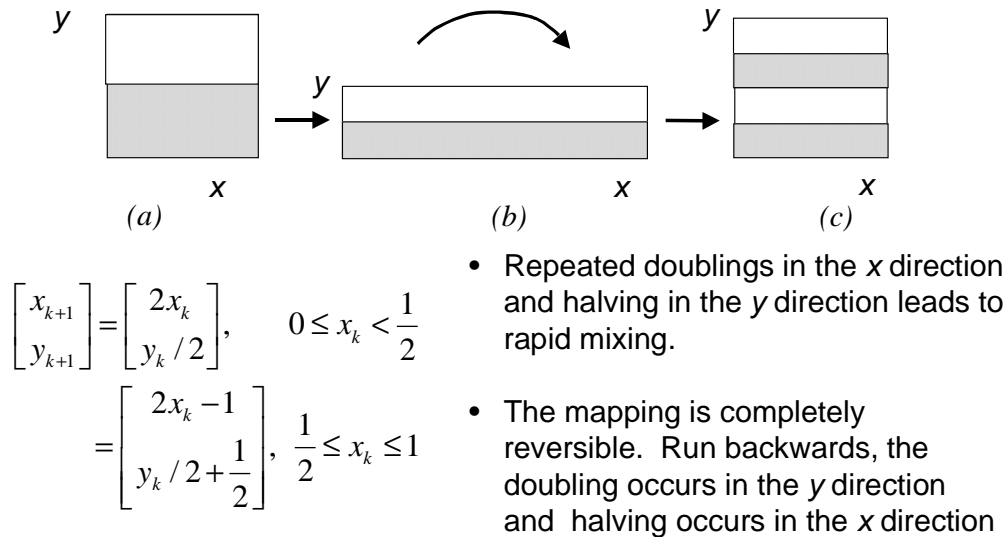
Figure III-24 Bakers Transformation

Research by Prigogine has also shown that irreversibility is linked only to Lyapunov time for general irreversible phenomena such as diffusion and various other transport processes (Prigogine 1997 p105). We thus have a link between these dynamical systems and technology transfer models herein. In Figure III-24, we observe that one direction *x* is expanding while the other dimension *y* is contracting. This is similar to our model where the amount of information that is discovered is equivalent to the amount

that is no longer undiscovered if the system is defined as two subsystems. Another view is to think of a part of the model that is restructuring the internal organization of existing information, and the addition of more information that is transported across the control boundary. After n consecutive iterations the distance between two points on the x will be multiplied by a factor $2^n = e^{n\ln 2}$. More will be said about this, however, according to others (Prigogine 1989 p254), Farmer (1983), (Baker 1990), we have a positive Lyapunov exponent.

$$\lambda_1 = \ln 2$$

This establishes the dynamic chaotic character of the system. Since this is a conservative system, the second Lyapunov exponent is negative $\lambda_1 = -\ln 2$. By repeating this process indicated in Figure III-24, which as time goes on each finite subregion will be partitioned into finer and finer strips. If some points (a representation of terms) were distributed as in a of the figure, we can see that after n iterations these terms would be diffused, mixed, in a number of ways.

Further discussion can be found in the appendix Appendix A Information, Control Theory and Evolutionary Dynamical Systems Basics, (p273) as well as in Prigogine (Prigogine 1983, 1989, 1997), Farmer, York Ott, (Farmer 1983), McCauley, (McCauley 1993), and Baker (Baker 1990). The following description follows the development found in Farmer (Farmer 1983) and Baker (Baker 1990).

So in Figure III-23, we see a plot of a one-dimensional map. Taking the derivative of $F(X_n)$ in this case yields λ. The goodness of fit is determined through the finite difference method. It defines convergence and stability points in dimensions using the Lyapunov number λ.

The Lyapunov numbers quantify the stability of an orbit around an attractor. The Lyapunov numbers are the absolute values of the eigenvalues of the Jacobian matrix at a fixed point. A discussion of the orbits, convergence, and stability for roots of different eigenvalues is covered in the appendix (Brown 2000, and Saboe 2001).

The eigenvalue of the characteristic equation $|A-jI|=0$, where A is the Jacobian of the transformation

$$\mathbf{TX}_{n+1} = F(\mathbf{X}_n) = \mathbf{TX}_n \qquad (3.43)$$

is

$$A = \frac{\partial(x,y)}{\partial(u,v)} = \begin{vmatrix} \dfrac{\partial x}{\partial u} & \dfrac{\partial x}{\partial v} \\ \dfrac{\partial y}{\partial u} & \dfrac{\partial y}{\partial v} \end{vmatrix} \qquad (3.44)$$

The Jacobean is defined by (3.44). The vectors $\mathbf{X}_{n+1}, \mathbf{TX}_n$ are defined in bold face characters. Other restrictions on (3.44) are that functions $x=x(u,v)$ and $y=y(u,v)$ have partial derivatives. For the point (x,y) corresponding to any (u,v) in R^* lies in R, and conversely to every point (x,y) in R there corresponds one and only one point (u,v) in R^*. (Kreyzsig 1993, p519-520).

The difference equations representing the dynamical system relationship to entropy through the Lyapunov number is defined as

$$J_n = [J(x_n)\ J(x_{n-1}).\ \ J(x_1)] \qquad (3.45)$$

where A is the Jacobean matrix of the map with $j_1(n) \geq j_2(n)... \geq j_p(n)$ are the magnitudes of the eigenvalues of J_n. A is the Jacobean matrix of transformation $\mathbf{T}$.

The Lyapunov numbers are

$$\lambda_i = \lim_{n\to\infty} [j_i(n)]^{1/n}, \ \ i = 1,2,...,p \qquad\qquad . (3.46)$$

The Lyapunov number is the smallest, positive, real *n*th root taken. We follow Farmer's assumption that *almost every* (Farmer's emphasis) initial condition in the basin of the attractor has the same Lyapunov numbers[19]. This followed from his empirical evidence, and the data in this model does not appear to meet the exceptional conditions that he identifies.

These dimensions represent an entropy measure for non-linear systems in stable or chaotic regions.

We compute entropy in two ways. One is from experimental data. The other is from a model of the process of transferring (transforming) information.   The experimental entropy data are related to content of a message, i.e. the information we know about a topic. We refer to this as Shannon's entropy (S_H). The data S_H is gathered over *k* time steps.

We perform regression analysis on this data and have therefore a function that is of the power function form. e.g. $y=bx^m$. This is

$$log\ y = log\ b + m\ log\ x \qquad\qquad (3.47)$$

where *m* is the slope and *log b* is the intercept in linear form. We also have a model of a non-linear dynamical system. The Lyapunov exponent of a map gives the sensitive dependence upon initial conditions that is characteristic of chaotic behavior. Further discussion can be found in Prigogine (Prigogine 1983), Farmer, York Ott, (Farmer 1983), McCauley, (McCauley 1993), and Baker (Baker 1990). The following

---

[19] The Lyapunov exponent is the logarithm of the Lyapunov number for the eigenvalues of the characteristic equation (Farmer 1983).

$$\ln \lambda = \lim_{n\to\infty} \frac{1}{n} \sum_{i=0}^{n-1} \ln|j_i^{(n)}|$$

description follows the development found in Farmer (Farmer 1983) and Baker (Baker 1990).

2. One Dimensional Finite Difference Representation of S_{H_k}

We determine the one-dimensional model for computation of this entropy for the *TechTx Basic Entropy* model in a form compatible with the two dimensional micro level model. This is

$$S_{H_{k+1}} = f(S_{H_k}) \qquad\qquad\qquad (3.48)$$

$$\lambda = f'(\bullet) \qquad\qquad\qquad (3.49)$$

The macro entropy is partitioned and allocated to the performer and affiliated organization nodes. This enables computation of the system entropy at the nodal level. This provides the method of computing the Lyapunov dimension from λ to measure the non-linear system entropy $S_{B_{micro}}$, at the micro level or for simplicity of notation, S_B. Note that this differs from the entropy S_H in Figure III-9, which is the information entropy, NOT the entropy measure for the stability or chaos of the system.

The general form for the transformation is $S_{H_{k+1}} = f(S_{H_k})$. We have from our earlier *TechTx Basic Entropy* discussion the macro entropy vs. time.

We develop the relationships using a power law here. However, as experimentation progressed, it became apparent for the technology we were evaluating that the messages were varying over time linearly and the entropy seemed to follow a power form.

As the power law may be the right fit for some technologies, we develop this more general relationship here. For the linear fit, the derivative reduces simply to a constant – the slope *m*. At the end of the day for the linear fit proved to be a very good and simple relation that gave most satisfactory results. While we recognize that we have to partition and allocate the entropy to the performing nodes, we can use the macro

function for illustrative purposes here. Having fit the entropy over time, we have a power function in the general form of $S_{H_k} = bk^m$

To derive the finite difference form, we have

$$S_{H_k} = bk^m$$

$$k = \left(\frac{S_{H_k}}{b}\right)^{\frac{1}{m}}$$ (3.50)

$$S_{H_{k+1}} = b(k+1)^m$$

Recall the general form of the finite difference transform is

$$S_{H_{k+1}} = f(S_{H_k})$$

To obtain the derivative, we use (3.50) eliminate k resulting in

$$S_{H_{k+1}} = b\left[\left(\frac{S_{H_k}}{b}\right)^{\frac{1}{m}} + 1\right]^m$$ (3.51)

To find λ we get

$$\lambda = \frac{dS_{H_{k+1}}}{dS_{H_k}} = \left[\left(\frac{S_{H_k}}{b}\right)^{\frac{1}{m}} + 1\right]^{m-1}\left(\frac{S_{H_k}}{b}\right)^{\frac{1}{m}-1}$$ (3.52)

Recall that λ was required to compute the Lyapunov dimension from λ to measure the non-linear system entropy, S_B to quantify the stability of the system.

3.   **Two Dimensional Finite Difference Representation of** S_{H_k}

Similarly, we develop a two dimensional model using the finite difference method. For n dimensional maps, there are n Lyapunov numbers λ_i, since stretching can occur for each axis.

A two dimensional model is used for the computation of the Lyapunov dimension from λ to measure the non-linear system entropy S_B.

$$
\begin{cases}
S_{H_{k+1}} = F(S_{H_k}, N_{i_k}) \\
N_{i_{k+1}} = G(S_{H_k}, N_{i_k})
\end{cases}
\tag{3.53}
$$

Functions F and G are defined as one-to-one functions in R. We assume that the partial derivatives exist. Now using λ as defined in (3.46) or (3.52) $\lambda = \lim_{n \to \infty} [j_i]^{\frac{1}{n}}$ where j_i are the eigenvalues of $|A - j_I I| = 0$ and A is the Jacobean of transformation is defined as $D\mathbf{T}$

$$
D\mathbf{T} = \frac{\partial (F, G)}{\partial (S, N)} = \begin{pmatrix}
\dfrac{\partial F}{\partial S_{H_{k+1}}} & \dfrac{\partial F}{\partial N_{i_{k+1}}} \\[2ex]
\dfrac{\partial G}{\partial S_{H_{k+1}}} & \dfrac{\partial G}{\partial N_{i_{k+1}}}
\end{pmatrix}
\tag{3.54}
$$

Here we are computing F and G to develop the transfer function and to correlate these two dimensions to determine S_B from λ, the Lyapunov number. The interesting feature of the bakers' transformation is that it is a dissipative function in state space since the sum of the exponents is negatives (Baker 1990 p122).

The entropy developed via discrete (micro) dynamical systems model and macro level computations both should change at the same rate since we are observing the same system. The performance index parameters are adjusted to tune the micro model and to match the S_B. This provides a method to identify the performance bands and half-life of performance improvement, or maturing of the technology.

4.   Micro Level Coupled Nodes Communicating

Let's give an example of information being exchanged at the micro level. Consider some coupled nodes in a communication system. This example is adapted from Brown (Brown 2000). This system described will be represented in a dynamical system model, which ends up being the bakers transformation.

This can be represented in a model of information and the state as it flows from the advocate and receptor as seen in Figure III-25. Model the following communication nodes, a *sender* (S), a *receiver* (R), and a *consumer (C)*. A simple function with inputs as messages and outputs as messages associated with each node carries the dynamical information about each node.
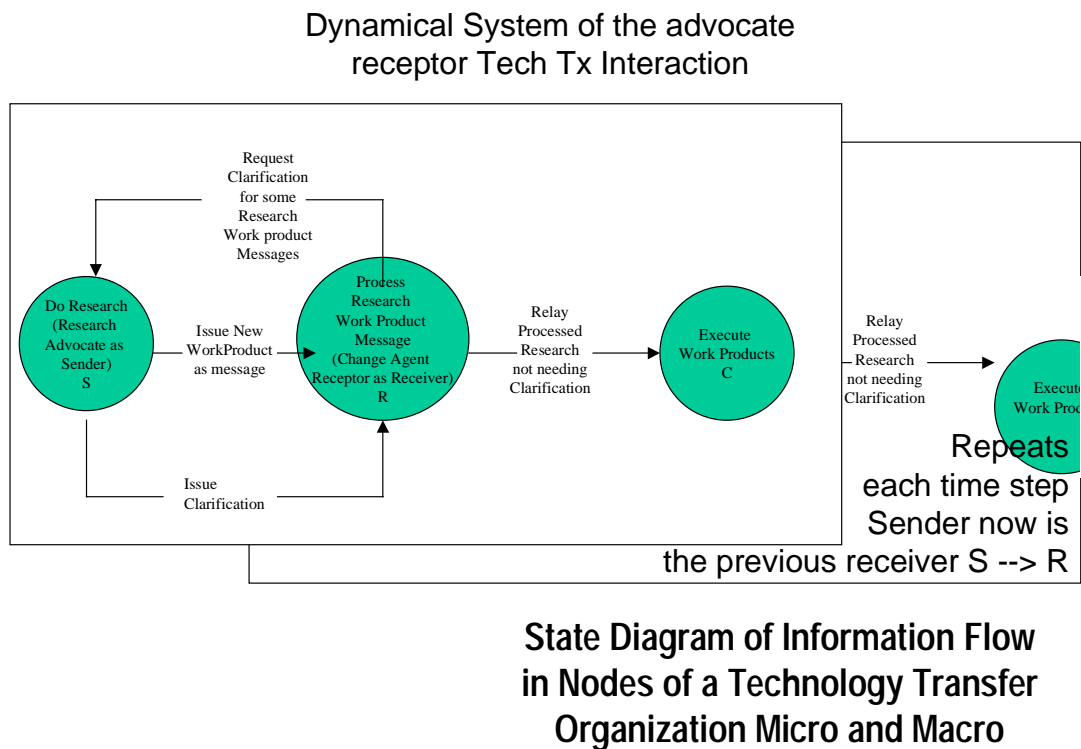
Figure III-25.   Dynamical System Model of Advocate-Receptor Interaction.

The sender is an advocate. This is a researcher, or in the terms of Fowler (Fowler 1994) an advocate and producer. The sender issues new work products as messages. The receiver is a change agent, or the receptor. The sender develops research, advances and publishes a message as a work product, thesis, article, technical report, demo, etc. The message is observable, e.g. measurable and countable. We can generally only measure output. We can measure output in terms of messages and terms from which the messages are made up. Except for one type of input, it is usually difficult to quantify, or measure all of the input.

The receiver receives the message. If the message is understood completely, i.e. no need for clarification, the receiver retransmits the processed message and a local state transition occurs on the node, as the receiver becomes a sender. The consumer node becomes a receiver, and so on, further down the technology transition food chain. On the other hand, if some percentage of the messages is not understood, the receiver asks for clarification in terms of feedback from the sender. The sender then sends clarification in response for the request for clarification. Another way to look at the request for clarification, is as a receptor, or researcher, we check the literature. The percentage of information we use is the complement to the request for clarification. The feedback gave us satisfactory answers. It becomes input from the world of persistent information available through time. This is the part of the world of information of sets of sets of terms (primitive messages) that the performer will restructure.

Once the consumer understands the message, the consumer can execute the work products. Since a change agent becomes a sender, and the consumer becomes a receptor, each is capable of issuing requests for clarification and providing clarification.

This elemental system (Figure III-26a) consists of a send unit and a receive unit. The receiver unit is able to retransmit or execute an action when there is little uncertainty in the terminal action to be taken. At that point, the receiver executes the action and becomes a send unit, since someone else (another potential receive unit) can witness the evidence of a signal. Let's assume for the moment a clear, noiseless signal from the sender. If the receive unit understands the encryption and protocol of the sender, it is

able instantaneously to resend the message or to act. No effort is required to handle the encryption and protocol.

If the message received is well understood, the unit R (at time step t_k) can receive the messages from unit S (sent at time step t_{k-1}), immediately and resends or performs an action, observable as a message, to another (or the same) receiver at a later time step (t_{k+1}). Figure III-26 shows this basic state transition model. Note, that there is also a term p' representing message state transition arcs for feedback. The message traffic *from* the receiver R is a sum of the fraction of messages from the earlier send units production and multiple streams persistent in history that are available to the receive node and selected (filtered) as input. The sum of the messages is available to be processed by node R.

5.   Entropy in the Communication Control Model

We can also have the case where there are messages with entropy (noise, or unknown signal) as input to R. This can be accommodated as seen in Figure III-26b. Now, we add the concept of a "think" state transition. This is the case where the messages received could not be effectively processed. Some internal processing is required. There is yet another type of "think" state transition. This is represented by feedback in order to clarify the entropy, noise or non signal received. Figure III-27 illustrates the elemental notion presented in Figure III-26b and adds two feedback loop state transition arcs p_4 and p_5. For initial model development and clarity, we assume that the quantity of messages in the think loop p_3 is equivalent to the number of messages sent back to the send unit in p_4. These are subsequently fed to a receive unit as clarification at some later time step as p_5. It is possible that the send unit has to use multiple time steps and its own think loop. Further, it is possible that the receive unit has to do more internal processing (and learning) which could store, for more than one time step, a number of prior messages awaiting action. We want to avoid or minimize a design that has this characteristic. The system would appear to have slow response to transients, and the hysterisis effects resulting from these time step delays can put the node and system in an unstable mode of operation. While some of this effect is unavoidable, the model should

be able to accommodate these aspects as well. We hide this essentially inside the nodes performance function. Refinements to this engineering model can be added later.

The nodes can be in two states, x_k, y_k in phase space. The state represented by variable y_k is the quantity of messages or tasks orders that have been executed by an organizational unit, or node at time t_k. The state x_k is the quantity of messages / task orders received by the organization at time t_k. x_k consists of two parts. One is the quantity of messages / task orders that the node adds to the system. In a sense ,new terms are added across the control boundary so they appear to arrive from the outside the organizational node. The second part is the set of internal messages / task orders that must be processed/executed by the unit due to the content of the messages / task orders processed in the previous time step (feedback) t_{k-1}.

Software Technology Transition Communications State Model "Basic" and with "think state"

a) Basic state transition -- interaction --
well understood effort
($p_2 = p_1 + p'_{technologies}$)

S p_1 R_k p_2 R_{k+1}

p'

b) Basic Interaction with a "think state" p_3
($p_2 = p_1 + p'_{technologies} - p_3$)

S p_1 R_k p_2 R_{k+1}

p_3

p'

S == send node state (of a unit), typical
of outside signal from earlier time steps
R == receive node state (of a unit),
with think and feedback states
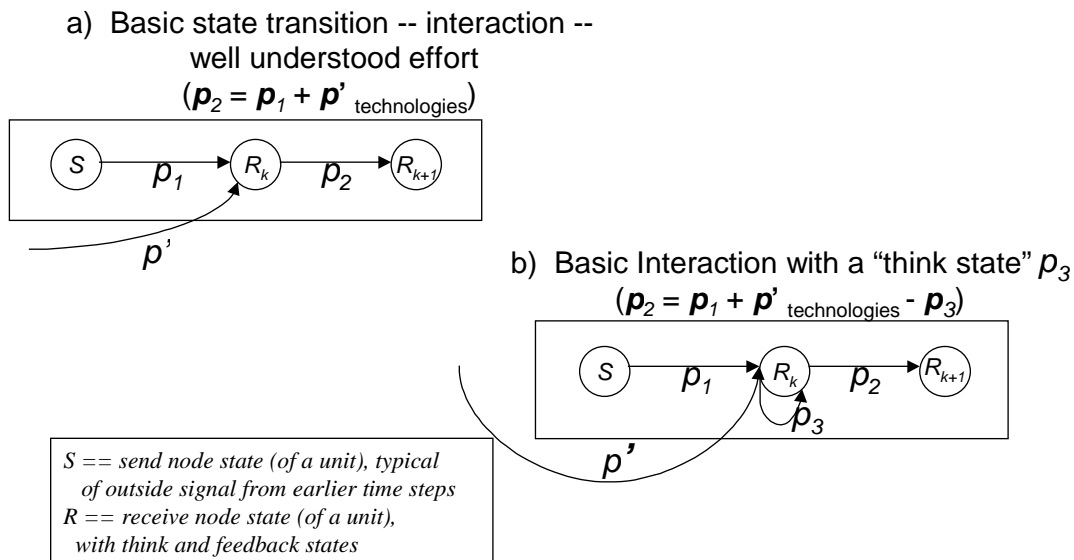
Figure III-26. Software Technology Transition Basic and "Think" State.

(Source: Saboe 2001)

Software Technology Transition Communications State Model "Think" and feedback

$$x_k, y_k$$

$$S == send\ node\ state\ (of\ a\ unit),\ typical$$
$$of\ outside\ signal\ from\ earlier\ time\ steps$$
$$R == receive\ node\ state\ (of\ a\ unit),$$
$$with\ think\ and\ feedback\ states$$
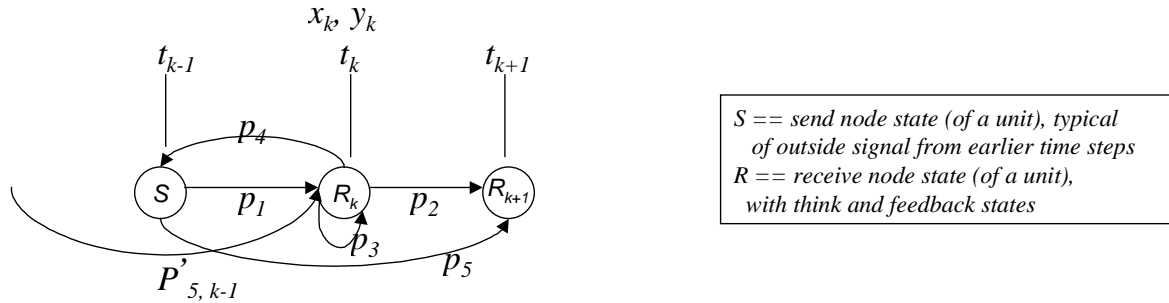
State variables:

p_i = *probability - property that must be conserved*

$x_k = u_k$ = *Quantity of Messages received from outside at time* t_k, $p_{1\ and}\ p_{5,\ k-1}$

y_k = *Quantity of Messages executed at time* t_k, p_2

$z_k = y$ *Quantity of Messages due to* t_{k-1} *clarification plus* x_k, p_1, u_k

$p_{4\ feedback} = p_{3\ internal\ processing}$ *at time* t_k

$p_{5\ clarification} = p_{4\ feedback}$ *at time* t_k *delayed by one time step* t_{k+1}

$P'_{5,\ k-1\ clarification} = \Sigma$ *all outstanding feedback messages from prior time steps that will be received as* x_k
and multiple streams persistent in history and available to the receive node which may be processed

Figure III-27. Software Technology Transition "think" and Feedback.

(Source: Saboe 2001)

On the other hand, let's assume that the receiver has to process some internal messages in order to unpack the message. Now there is a delay before the message can be resent. Going a little further, if the receiver received noise, an unclear signal, or unknown signal it may have to request clarification, delaying a time step or do some additional correction processing. This uses up node capacity. We know from experience, that when we are fully consumed with a project, day and night, we are not available for other tasks. This capacity can even limit interaction with the external environment (e.g. in extreme cases, this is capacity can even be unavailable for the researcher's family). If the message is simple and concrete, or agrees in abstraction (state level) or is at a higher level meta-statement, the amount of processing and effort that it takes to correct the poor signal is less than one that is more complicated and more densely packed. From this, we

might say that abstraction is a form of information hiding. Encapsulation of this form provides leverage and can reduce the "entropy" of the system. The complexity of the structure of the message is higher, but the communication is using less bandwidth.

D.   DYNAMICAL SYSTEMS MODEL

Assume we have available a macro level model of technology transfer to represent the community level technology maturation. That macro model can identify the stability and convergence of an ensemble of nodes. The macro model can be partitioned into a number of nodes (organizational units and sub units that compose the organizational units). The macro model is represented in terms of entropy dimensions of natural measure (Farmer 1983), i.e. both the information entropy S_H and the bakers' transformation entropy S_B, representing the transfer (transform) function. We now would like to develop a model that represents the interaction between nodes at the micro level. This model will complete a linkage from macro to micro levels and permit implementation models (infusion, learning, etc.) to bridge to the macro-micro infrastructure scale models. This section will explore a feedback model at the organizational node and sub-organizational node level. We incorporate control theory and use the bakers' transformation.

The model should incorporate a factor for learning, and address requests for clarification and the ability to model the process load in requesting clarification messages and receiving clarification messages. This model will permit tuning an organization to ensure efficient processing of technology messages. We will develop a node response curve and associated system response curve these can be developed from the macroscopic view. Determination of the bakers' transformation entropy from the Lyapunov number and exponent will permit an assessment of the node performance in terms of stability and confidence of convergence to a steady stable state, or chaotic state.

1.   Assumptions

Assume nodes made up of people and machines that can do a task, such as publish a work product as a message. A node is modeled in terms of the messages it receives verses the messages it processes. The work product (message) is the representation of something that can be understood by communicating in terms familiar to the sender and receiver. For instance, a map is not the road system but symbols from a vocabulary of terms that represent a common understanding of the lay of the land of a road system. The terms are measured in information units – bits. As input, the processing node receives work product. These represent messages. Output from a node is also observed and measured in messages. A technology generating or processing node produces the output by acting on input to reduce uncertainty in the cause and effect relationship involved in achieving a desired result. This is reasonable since this is what elements of a node do. This is true for the activities of researchers, producers in general as advocates, or receivers, change-agent and consumers as receptors. This assumption is also consistent with the observation by Rogers (Rogers 1983). Within this context, we examine the meaning of the concepts of stability, equilibrium, attractors, chaos, eigenvalues, and eigenvectors, and the relationship to technology transition and, system node dynamics. Convergence of an organizational node on a fixed point depends on the nature of the eigenvalues of the derivative of the dynamical system at the fixed point. The direction of convergence depends on the direction of the eigenvectors. A useful term that will frequently arise in the following discussion is an "orbit'. The orbit of a dynamical system is that sequence of points in the state-space phase plane that corresponds to successive time steps in the system. We discuss seven cases in the appendix.

2.   Context

We assume that all of the nodes have functions of equivalent form. As described in the *TechTx Entropy Learning Curve* model, nodes, in different performance bands, inherit the performance parameters of their band. The node is modeled in terms of the messages it receives versus those it carries out or processes. The individual nodes are assumed heterogeneous, varying in size and composition, or a mix of people with varying

skills and tools to perform the function. For ease in validation computations, we assume that of the organizational nodes that have a performance index in the range of +/- 1σ of the mean (recall Figure III-21), all would have the same learning curve function parameters. We can partition the volume down into finer and finer bins. The best model would look at all of the sets of sets of performer combinations and partition this into q-levels. For now, however, it suffices to allocate the nodes with statistically similar performance to one of the four appropriate bins.

Should we wish to calibrate an individual node or all of the nodes in the band, the model will still be applicable. The capacity of a node in the band can be calculated. The volume and complexity of messages acted on and generated applies pressure to an organizational node. Demands on the organizational node as a sender or receiver component are among the pressures that require modeling and analysis. Other pressures are internal to an organizational node to ensure smooth functioning. These internal pressures come in the form of messages as well, and procedures, interfaces, meetings, collaborations and other interactions that consume resources. These are important facets to model since they provide feedback pressures on the components. External pressures are also among the features that determine organizational node dynamics and this should be modeled.

All of the pressures mentioned so far can be thought of as messages passing between organizational nodes and between the organizational nodes and the environment. This concept facilitates modeling organizational node states that can be organized as messages received by the component and processed by a component. In this respect, the organizational nodes are analogous to a communications network. The analog is simple and useful. There are, however, at least two important differences. One is that an organization will adapt to and absorb pressures that would cause a network to breakdown. This is because the network is not hardwired. It is also difficult to predict the breakdown capacity in advance. We have somewhat addressed ranges of capacity by banding the organization into performance index bands. This however does not mean that a node is at capacity. The potential for the technology transfer system to break down is important to model. A simple source of collapse is when the demands on the system exceed its ability

to adapt, and the node reaches a state of demoralization[20]. This is important since it can result in a component ceasing to communicate, or the communications decreasing to a critical level. In the communications network analogy, the number of messages being processed begins to decay until it reaches an inoperable level or is zero. We have mechanisms to model this, however for purposes of illustrating the model, are at or below capacity. We can ignore for now this breakdown at over capacity issue.

The model for organizational dynamics is drawn from (Brown 2000). This model can be represented in state space using the messages (N, and primitive messages term sets of sets (n). This can be related to entropy (S_H). For notation ease, we drop the subscript indicating that this is entropy in the terms of Shannon.

The state space is mapped onto the x-axis (input) and y-axis (output) as follows: x, the input N_k, in messages represented as entropy in information units, and the output in y, N_{k+1}, where k represents the time step. (The internal time as an *operator*, and not a number). We would not have synchronous discrete time steps in a network that includes nodes comprised of organizations and people.

$$N_{k+1} = f(N_k) \hspace{3cm} (3.55)$$

This function represents the bakers' transformation. For the ensemble of nodes performing the function $N_{k+1} = f(N_k)$, we have the vector representation $N_{k+1} = F(N_k)$.

We narrow our discussion from the ensemble of messages operator on by nodes, which appear on the network or disappear to a typical group of nodes: the sender, receiver and consumer.

The model uses two state variables. A variable of the system node representing the messages received and one for messages processed. We shall apply the message information in terms of the entropies of the incoming and processed messages. The significance of the system of equations is that the eigenfunction characteristic equation represents the bakers' transformation of folding, stretching and rotating. The eigenvalue

---

[20] The overheating of the internet dot com start ups is an example of organizational nodes that were under too much pressure. Competing at "internet time" caused many organizational burn out tragedies.

of this dissipative function is also entropy, and it represents mixing. The appendix examines a number of cases and discusses the potential significance of the values of the eigenfunction.

Prigogine (Prigogine 1989 p198) summarizes of how the general properties of a dissipative dynamical system can be represented and evolves. He states that the very existence of dissipative dynamical systems is a manifestation of the second law of thermodynamics.

3. Dynamical Systems Model Equations

Now we will develop the equations for this model. The relationship between the state transition diagram and a dynamical system is shown in Figure III-28. The sender publishes messages u_k (a natural number of messages) at time step k. Input messages at time step k to the receiver are indicated by x_k (a natural number of messages). The output messages from the receiver at time step k are given by y_k. (a natural number of messages) Some percentage of the messages' output from a prior time step y_{k-1}, are indicated by β, a rational number.

This process is repeated for the next time step x_{k+1} and y_{k+1}. The crossed circle immediately to the left of the receiver node represents the collection point where the different parts of the input message stream are combined for the input message count x_k.

In Figure III-28, $f(x_k)$ represents the function to transform the input messages into output messages. It takes a time step to complete the processing. A way to view the nodes processing is that for a message to move through a node, it takes a time step.
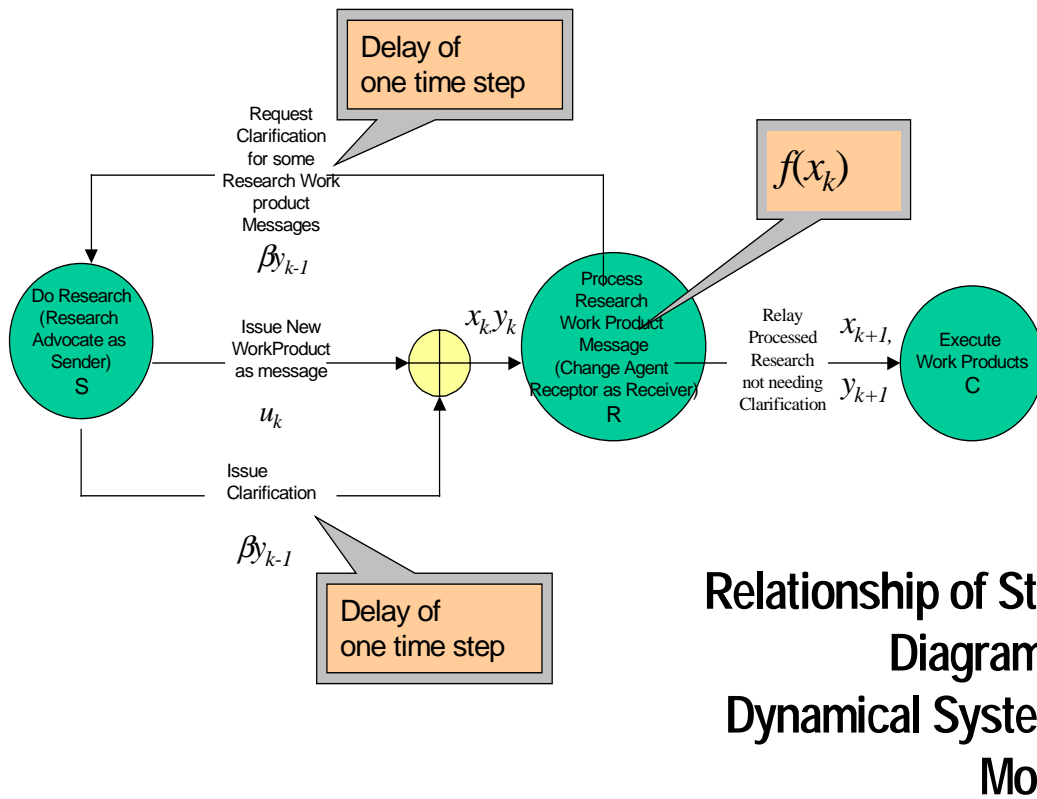
Figure III-28. Dynamical Systems Model.

The x_k state variable consists of two parts. One part of the state is the messages that come from outside the receiver node u_k i.e. from the sender node. These are new messages consisting of terms that count be either the conversion of questions {???} to answers {!!!} from term sets that were previously nulls {}. Alternatively, the answers {!!!} that may have been previously discovered, which contribute to more mutual information. The second part of the state variable is clarification of messages that was requested from the previous time step y_{k-1}. Initially we assume that the quantity of messages processed (y_k) is a function of x_k. As we said earlier, while it may appear that we could have non-determinism here, this is not the case. If we could know all of inputs, there is a deterministic relationship, however, it impossible to know all of the inputs. We

- 153 -

must distinguish between non-deterministic and probabilistic. We simple don't have enough information to accurately predict the result. A function could be a reasonable approach. . This function has the following properties:

(1) if $x_k = 0$ then $y_{k+1} = 0$
      and
(2) as $x_k \to \infty$ then $y_k \to 0$

Condition (1) says if there is no input at time step k, there is no output. This holds only if there are no messages stuck in the node or latent messages in the form of clarification coming in from prior execution steps. Condition (2) says that the system grinds to a halt if the message demand is too great. We can assume that as the number of messages received becomes infinite, the messages processed have to approach some limiting value, which is the capacity of the system. The system can be represented by the following equations. However, for the systems we are seeing, we are not at capacity, and this condition can be finessed out of the picture in low pressure, low temperature situations. We can determine when this happens by partitioning the macroscopic community into smaller and smaller partitions. Then we can observe the performance of nodes with a technology and in the environment of the day.

$$x_{k+1} = \beta y_{k-1} + u_k$$
$$y_{k+1} = f(x_k)$$

(3.56)

$f(x_k)$ is called the node response curve. We need only concern ourselves, for this exposition, on the node response curve and its ultimate relationship to the macroscopic information theoretic model. The above is a second order system of finite difference equations with the response curve $y_{k+1} = f(x_k)$ represented by the following three-dimensional dynamical system. Where z_k, clarification from the prior time step, is substituted for y_{k-1} and using the mapping referred to in (3.43) and (3.44) * MERGEFORMAT (Kreyszig 1993 p419) we get

$$\begin{pmatrix} x_{k+1} \\ y_{k+1} \\ z_{k+1} \end{pmatrix} = \mathbf{T} \begin{pmatrix} x_k \\ y_k \\ z_k \end{pmatrix} = \begin{pmatrix} \beta z_k + u_k \\ f(x_k) \\ y_k \end{pmatrix}$$

(3.57)

Let's assume X,Y,Z is the time step $k+1$.

The periodic points determine the dynamics of the system. In particular, the fixed points are of interest. These are the equilibrium points. The coordinates of the fixed points are given by

$$\begin{pmatrix} X \\ Y \\ Z \end{pmatrix} = \begin{pmatrix} \beta z_k + u_k \\ f(x_k) \\ y_k \end{pmatrix}$$

(3.58)

The fixed-point condition becomes $x = u + \beta f(x)$. The derivative of the transformation $\mathbf{T}$ is given by the Jacobian

$$J = \frac{\partial(X,Y,Z)}{\partial(x,y,z)} \tag{3.59}$$

$$= \begin{vmatrix} \dfrac{\partial X}{\partial x} & \dfrac{\partial X}{\partial y} & \dfrac{\partial X}{\partial z} \\[2ex] \dfrac{\partial Y}{\partial x} & \dfrac{\partial Y}{\partial y} & \dfrac{\partial Y}{\partial z} \\[2ex] \dfrac{\partial Z}{\partial x} & \dfrac{\partial Z}{\partial y} & \dfrac{\partial Z}{\partial z} \end{vmatrix} \tag{3.60}$$

$$D\mathbf{T} = \begin{pmatrix} 0 & 0 & \beta \\ f'(x) & 0 & 0 \\ 0 & 1 & 0 \end{pmatrix} \tag{3.61}$$

Where $D\mathbf{T}$ is the Jacobean of transformation $\mathbf{T}$. Find the eigenvalues j_i which are the roots of the characteristic equation:

$$\left| A - jI \right| = 0 \tag{3.62}$$

Where A is the Jacobian of the $D\mathbf{T}$ transformation, and I is the identity matrix. More specifically the determinant

$$\left| D\mathbf{T} - jI \right| \tag{3.63}$$

is the characteristic equation when set equal to zero.

$$-j^3 + \beta f'(x) = 0 \tag{3.64}$$

There are three eigenvalues for the solutions of the equation $j^3 = \beta f'(x)$. There are two complex conjugate eigenvalues and one real eigenvalue. The three eigenvalues may be represented as

$$j_1, j_2 e^{(2\pi i/3)}, j_3 e^{(-2\pi i/3)}$$
where
$$j_i = (\beta f'(x))^{1/3} \tag{3.65}$$

is the real root of the equation. From the model, we conclude that $\left| j_i^{\,3} \right| < \beta \left| f\,'(x) \right|$.

The system is stable when $\left| j_i \right| < 1$, in equilibrium when the norm is $\left| j_i \right| = 1$, and unstable when $\left| j_i \right| > 1$ (Farmer 1983, Baker 1990, Brown 2000).

This gives some insight into the structural stability aspect. The control theory element of the current research model addresses mixing, and structural changes due to feedback from external nodes. The value of the norm (<1, =1, >1, real imaginary, etc) of the eigenfunction characteristic equation assimilation of reality based on experiences from prior time steps.

From this, we see that for small enough β or large enough u_0 we can achieve stability. For the technology transition system, we desire stability and convergence. With a stable model at the organizational level, we have organization nodes, which are not thrashing or wasting effort. With stable nodes, we can build a stable infrastructure composed of those nodes. This will also yield convergence of the technology.

The data that we can measure is the number of messages published at some time k. We can also measure output $y_{k+2,\ k+1,k,\ k\text{-}1,\ k\text{-}2,}$. The output message data is simply the offset published by a time step e.g. $u_{(t\text{-}c)}$. The difficulty we have is, that the macro data to empirically support $f\,'(x)$ cannot be arrived at directly.

Our system curve from empirical data is the output y, which represents u offset by an interval c from a prior time step. Initially, for the data examined, this interval was one year. In effect, this provides an immediate memory for chunking of three registers because it take three time steps to clear all of a message when there is a request for clarification.

As this immediate memory, represented in time steps is expanded, the error from the modeled to predicted should start to diminish. Therefore:

$$\mathbf{Y} = y_t \equiv u_{(t-c)} \tag{3.66}$$

$$\mathbf{X} = \beta f(x) + u_t = \beta u_{t-c} + u_t \tag{3.67}$$

Now deriving from (3.66) and (3.67) we get

$$f'(x) = \frac{d\mathbf{Y}}{d\mathbf{X}} = \frac{d\mathbf{Y}/dt}{d\mathbf{X}/dt} \qquad (3.68)$$

The following result was obtained using parametric differentiation of (3.68) and substituting (3.66) and (3.67).

$$f'(x) = \frac{u'_{(t-c)}}{\beta u'_{(t-c)} + u'_{(t)}} \qquad (3.69)$$

We can substitute $f'(x)$ into (3.65) which defines the real eigenvalue:

$$j = \left(\beta \frac{u'_{(t-c)}}{\beta u'_{(t-c)} + u'_{(t)}} \right)^{1/3} \qquad (3.70)$$

or explicitly to enable programming from the data sets

$$f'(x) = \frac{\dfrac{du_{(t-c)}}{dt}}{\beta \dfrac{du_{(t-c)}}{dt} + \dfrac{du_{(t)}}{dt}} \qquad (3.71)$$

The point where the graph intersects the line $y=x$ is the equilibrium point. The slope of $y=u+\beta f(x)$ at the fixed point is the real eigenvalue of the matrix $DT(X)$. By changing the parameter β, we change the shape of the graph and thus we change the slope where the fixed point is found. Also by changing u_0, we change the location of the fixed point along the horizontal axis and thus the eigenvalue. By starting u_0 at 0, we first have a fixed point whose real eigenvalue is positive and less than 1. This is ideal in that it indicates that the solution will converge to a point where it remains stable and makes sense. The review of the various characteristics of the eigenvalue is developed in the appendix. See these graphs and the various interpretations of their meaning in the appendix.

For the moment, let's go back to the model consisting of sender, receiver and consumer Figure III-26. Now let's focus in on the receiver and look at the inputs and

outputs of this node. It turns out that any of these nodes looks like a receiver in the general sense. The sender can also be picking up new messages from others, in which case the sender acts like a receiver. The sender can also be requesting clarification and be receiving clarification in the same manner as the receiver. Likewise, the consumer gets input and outputs. So our model can be seen in Figure III-29 to have all of the features but represented only in a single node, the receiver. When the "receiver" conjures up a goal set of objective terms of previously unanswered terms {???} and puts them into answers {!!!} in the system for the first time, these terms represent u_k, or the conditional probability P(Y|X) and conditional entropy $S(Y|X)$. The mutual information represents the terms that were previously know to the community, but were now reinforced with additional instances of the terms. Using the single node version of the model, we also have a useful sign convention. All of the inputs to the node are positive and outputs are negative.

Software Technology Transition
General Node Inputs and Output

S == send node state (of a unit), typical of outside signal from earlier time steps
R == receive node state (of a unit), with think and feedback state transitions
C == is a consumer node (state of a unit)

• p_1 is the input from a new publisher in this time step

• p_2 is the output, the publication in the time step

• p_3 and p_3' are memory in and out

• p_4 and p_4' are requests for feedback out of and into the node respectively

• p_5 and p_5' are responses to clarification requests out of and into the node respectively

The information is conserved in and out of the node (note sign convention) -- so
$$p_2 + p_3 + p_4 + p_5 = p_1 + p_3' + p_4' + p_5'$$

Figure III-29. General Node Inputs and Outputs.

We are now in a position to think of an ensemble of nodes. Essentially a distribution of these nodes is performing the bakers' transformation. Just like a physical system or communication system, we now can speak of a macro stochastic process in terms of entropy and information.

With the compelling evidence of the curve fit data in Figure IV-6, we reevaluated the eigenvalue function of the control equations using linear curve fit for messages verses time step.

$$u_{(t-c)} = mt + b \qquad\qquad (3.72)$$

where we are computing, the messages added to the system at time step t. Since the equation is linear, the more general form previously developed for a non-linear $u(t)$ to enable varying the interval over timestamp t-c has no effect on the additional messages added to the system in a timestamp. For our first approximation, the derivative of u_t will always be a constant. That is the slope m.

Then taking the derivative $du_{(t-c)}/dt$ in (3.72), we have a constant for j in (3.70). At this point we wish to tune β to see if the determine if the dynamical control model stabilizes with a function in the same form as the macroscopic entropy S_H.

We are now in a position to think of an ensemble of nodes essentially a distribution of these nodes performing the bakers' transformation. Just like a physical system or communication system, we now can speak of a macro stochastic process in terms of entropy and information.

Let's go to the basic equation (3.25). Recall our conserved property is messages N, information in our case. Using Shannon's entropy S_H and N for the number of messages we get

$$\frac{1}{T_{S_H}} = \frac{\partial S_H}{\partial N} \qquad\qquad (3.73)$$

From the section dealing with the information theoretic aspects of interaction subsystems, we saw how T varied with a time step. Now we can observe the control model, dynamical entropy, S_B, as a function of the same time steps. We have the

opportunity to relate the two entropy measures, S_H and S_B since they are related to the same information system of messages N. We are dealing with the same information flows, hence the same system, so this seems reasonable. Recall S_B is related to Lyapunov's exponent λ, which comes from the eigenvalue j.

We found the relationship of messages verse time step in Figure III-8 was very satisfactorily modeled as a linear equation for this technology set. (It could be different for other technologies, this is why we have dealt with the relationships in terms of functions, eigenvalues and derivatives.) In this case, the derivative of the linear model reduced to a constant in equation (3.72) as noted earlier.

Now instead of using an average, or guess for β, it is computed directly. To compute β, the amount of information that a node consumes which persists in time, both equations are a function of timestamp, so we can solve $S_B(k)=j_k$, $j_k = b_j k^{m_j}$ and $S_H(k)$ $S_H = b_S k^{m_S}$, for k. Setting them both equal to each other, we can solve for β (S_B, S_H).

$$\left(\frac{S_k}{b_S}\right)^{\frac{1}{m_S}} = k \qquad\qquad (3.74)$$

and

$$\left(\frac{j_k}{b_j}\right)^{\frac{1}{m_j}} = k \qquad\qquad (3.75)$$

which yields

$$j_k(S_k) = b_j\left(\frac{S_k}{b_S}\right)^{\frac{m_j}{m_S}} \qquad\qquad (3.76)$$

$$S_k(j_k) = b_S\left(\frac{j_k}{b_j}\right)^{\frac{m_S}{m_j}} \qquad\qquad (3.77)$$

Here the subscripts s refers to slope and intercept terms of the Shannon entropy equation, and the subscript j is referring to similar terms in the S_B, bakers transform equation.

Before we do, let's explore the relationship to temperature from the discrete, micro, model. Earlier, using a macroscopic approach, we showed that temperature increases, or decreases with increasing or decreasing pressure on a node respectively. In a physical system, we can address temperature in of entropy and conserved property, let's see that this is true for this discrete, micro formulation as well.

In Figure III-30, we see on the left-hand side, that there is a transfer function that converts X input, or some percentage of the available persistent input, into Y, output. This is really made up of two parts as seen on the right. We can use a Venn diagram as introduced earlier. Extensive properties like messages are additive. Probabilities are multiplicative. This also applies to the entropy.

Node Input and Output
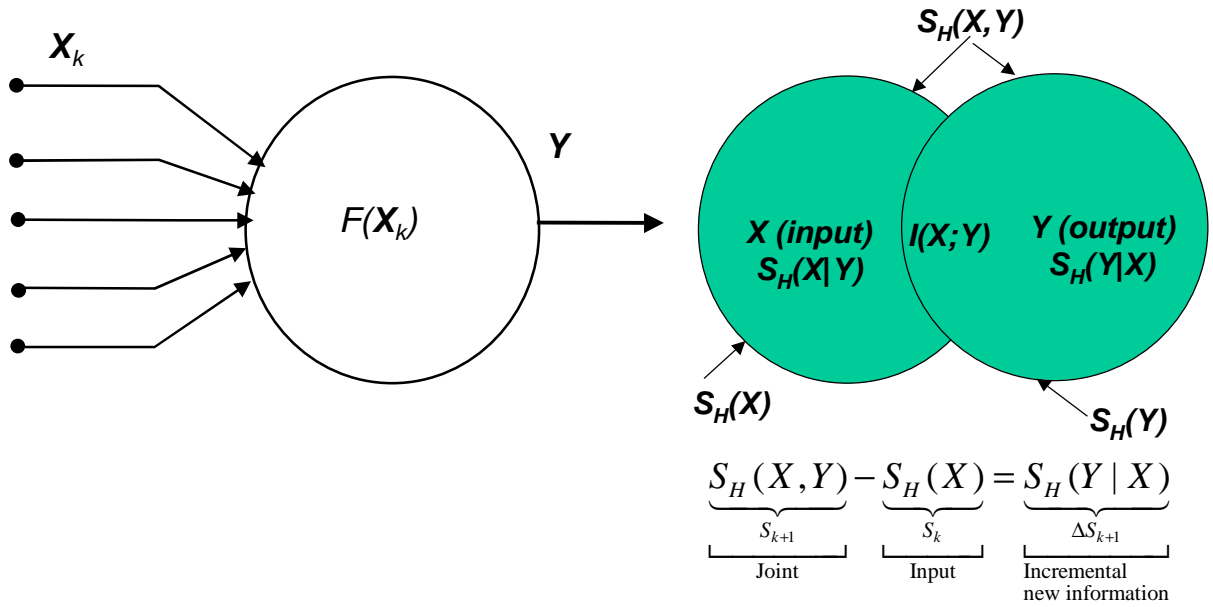
Figure III-30 Node Input and Output in terms of Entropy, and incremental new information

4.   Temperature from Discrete Control Model

Recall equation (3.1), now that we have a discrete model, we can develop a relationship that is consistent with entropic approaches of statistical mechanics.

Let's assume that we need to line up with the equation (3.25), we see (3.1), can be written as

$$S_{k+1} = S_k + m_N N_k \qquad (3.78)$$

$$N_{k+1} = N_k + m_S S_k \qquad (3.79)$$

Here we are suggesting that there is a linear (m) relationship of messages ($m_N N_k$) in (3.78) that are discovered, and added, and similarly that there is an entropy relationship

(m_S S_k) in (3.79). The respective m for the messages and entropy also will address the units.

Rearranging (3.78) and (3.79) we get

$$\frac{S_{k+1} - S_k}{N_{k+1} - N_k} = \frac{m_N N_k}{m_S S_k} = \frac{1}{\dfrac{m_S S_k}{m_N N_k}} = \frac{1}{T} = \frac{\Delta S}{\Delta N} \tag{3.80}$$

This makes (3.1) consistent with the definition given in (3.25) for temperature T. We can now write (3.78) and (3.79) in terms of temperature, since

$$T = \frac{m_s S_k}{m_N N_k} \tag{3.81}$$

$$m_N = \frac{m_s S_k}{T N_k} = m_N(S_k, N_k) \tag{3.82}$$

$$m_s = \frac{T m_N N_k}{S_k} = m_s(S_k, N_k) \tag{3.83}$$

These equations are nicely linked through temperature.

$$\overbrace{S_{k+1}}^{x_{k+1}} = \overbrace{S_k}^{\beta y_{k-1}} + \overbrace{\frac{m_s}{T} S_k}^{u_k} \tag{3.84}$$

$$N_{k+1} = N_k + T m_N N_k \tag{3.85}$$

This can be written

$$S_{k+1} = S_k \left(1 + \frac{m_s}{T}\right) \text{ and } N_{k+1} = N_k \left(1 + T m_N\right), \text{ however, examining (3.84) we can}$$

now better see the relationship to the feedback control model shown in (3.56). We will elect to have only one tuning parameter β on the entropy equation this time. This way we can relate the system to maximum entropy. Since the pair of equations are linked via their second terms, one tuning parameter is simple and sufficient.

Now we relate to the equations to the Venn diagram. The mutual information component $I(X;Y)$ is the part that deals with irreversible entropy, and $S_H(Y/X)$ represents a reversible entropy. The term βy_{k-1} is a way of looking at how much as a percent of the previous body of knowledge the research node is questioning and restructuring. In the control model the receiver node, asked for feedback from the early senders. The counterpart to the request for feedback was received as clarification. When the receiver node reaches out and touches the exiting structure of terms in various microstates, this is the percentage of the Venn diagram represented as mutual information in this time step. The entire Venn diagram becomes the contribution at this time step to the body of knowledge, along will the rest of the communities contribution at this timestep. This then is available as input y_{k-1}, or $S_{k+1}(X)$ and $PN_{k+1}(X)$ or the next time step.

Here we can look at the total entropy consisting of two parts as seen in Table III-1 and (3.86).

Irreversible $dS(irrev)$	Mutual Information $I(X;Y)$	Production βy_{k-1}
Reversible $dS(rev)$	Conditional Entropy $S_H(Y/X)$	Portation u_k

Table III-1 Model components

$$dS_{total} = \underbrace{dS(irrev)}_{\underbrace{I(X;Y)}_{production}} + \underbrace{dS(rev)}_{\underbrace{S(Y|X)}_{portation}} \qquad (3.86)$$

The im*port* or ex*port* of an extensive variable might be referred to as "portation". This is a change in entropy due to the addition or removal of some of the systems extensive property. This is when we add extensive properties across the control volume boundary and thus increase the bounded control volume. An example of this would be adding terms to the vocabulary. Then a researcher provides input in the variable u_k. This is, in principle, reversible. Another example, since we now have a relationship if extensive and intensive variables through temperature, might be adding or subtracting

volume, i.e. more nodes, organizations, authors, etc. This is what can be considered a "becoming" property.

The production component of the equation deals with the organization, or rearrangement of free or available microstates. The mutual information is dealing with part of the systems entropy that is locked up in the structure of the system. This is a function of the present organization of the microstates of the primitive terms. When a researcher moves and combines existing terms in an arrangement that was not previously populated, we are dealing with a "being" property.

These free states are those that are available for any system to change its future organization by conversion into chaos (usually heat) or order (usually work). These two components have an important relationship to the main differences between a learning organization and knowledge management, which are also typical of science, research and technology advancements. Hence, we have relevance to technology transition. In first and second-generation knowledge management, and technology transition, the focus is on the porting of knowledge, which happens reversibly. This is typical of traditional education. In a learning organization, the focus is on the production of knowledge that happens irreversibly. This is typical of competency based education and self-actualization (Maslov's highest level) which is done during advancement of science as in a effort to achieve a Ph.D. Here we are concerned with using the universal availability of free energy[21].

5.   Temperature and the Partition Function

We saw how microstates of an alphabet were related to entropy on p99. Look at the maximum entropy for a number of small alphabets.   indicates the number of potential microstates at each q-level for an alphabet of 128 terms. This is the peak in the middle, centered at q-level 64. Read the microstates for this curve only on the left-hand y-axis.

---

[21] Implications of an natural evolution, entropy always increases, toward stringing concepts to signed numbers so that the more complex our conceptualization becomes, the less our confusion (complication) becomes. We are moving toward knowing and being. This paragraph was the resulted from a chat room discussion on the web.

Temperature is related to the free or available microstates relative to the maximum entropy of the vocabulary.

Regardless, we can still look at the free "energy" (i.e. conserved property) states, that, is the available states to which terms could populate. Recall (2.9) where U represents the internal structure. Chemists actually have this well figured out. Thanks to Gibbs, they view components (our messages) each having a definite structure allowing a definite reaction mechanism. The work with $\Delta F = Q + W + ? + ?? + ??? + ...$ and a machine view is $\Delta E = Q + W + ? + ?? + ??? + ...$ where the original construct allowed for adding any yet undiscovered method of converting energy[22].

If the internal structure has available free microstates, we can stimulate the system to populate various q-levels with sets of sets. Then we can use the partition function related to the microstates of each q-level to determine the temperature.

This is very convenient since this is the partition function, the most useful equation is statistical mechanics and it contains the temperature term we desire.

$$P(q_i) = Ce^{-q_i/kT} \quad \text{Boltzmann-Gibbs} \quad (3.87)$$

If we sum all of the q-levels, we get Q_I given by

$$\Omega_c = \sum_{i \in \Omega_V} e^{-\frac{q_i}{kT}} \quad \text{Partition function} \quad (3.88)$$

where q_i is the property to be conserved, T is a temperature, k is a constant for unit conversion, and C is a normalizing constant. It turns out that the normalizing constant $C=1/T$. To get the units right for the conserved property with a constant volume $T=n/V$ or messages per node. It would appear that this conserved property can be records (a rich message), message primitives with a single term distribution, or message primitives distributed in sets of sets, or q-levels.

---

[22] We know that magnetic fields, and radiation fields at one time were not understood to influence the conversion of free microstates. Could we possibly someday see an information field theory, based on statistical mechanics, information theory, and *software physics*?

Here is how we relate this to messages and nodes. Using Boltzmann-Gibbs, we know that q_i is the sum of the primitive messages (n) in the bins.

We know that the total number of messages n are distributed over V nodes (authors). So following Boltzmann's logic, if this were continuous, we would have q_i distributed over an infinitely small size *(dq)* of infinite bins and we would have

$$\int_0^\infty P(q_i)dq = 1 \qquad (3.89)$$

and from the normalization condition which takes all of primitive messages $\Omega = n$ distributed over all of the nodes V

$$\int_0^\infty qP(q_i)dq = \Omega/V = n/V \qquad (3.90)$$

we find that

$$C = 1/T \qquad (3.91)$$

and

$$T = \Omega/V \qquad (3.92)$$

where $\Omega = n$ primitive messages (sets of sets of terms) We don't have an infinite number of bins, rather our bins are countable and numbered 1 through $|\tau|$, where $\tau \equiv \{terms\}$ and $2^{|\tau|} \equiv \{messages\}$

We can also maximize the entropy when there is an equal distribution of all of the terms $S = -\sum p(q)\log_2 p(q)$ which is

$$S_{max} = -\log_2 \frac{1}{2^{|\tau|}} \qquad (3.93)$$

This is the analog to *free* energy in statistical physics.

$$F = -kT \ln \Omega \qquad (3.94)$$

This is the available, or yet unoccupied microstate, not already tied up in the structure.

Depending on which type of message is chosen, you will get a different value for the conjugate Legendre developed intensive variable. Each will give a different temperature. The true temperature in fundamental information units *must* be done on the basis of the n-tuple pair-wise sets of sets combinations, which are allocated to q-levels.

Since each granularity of message has a different temperature term $\beta=kT$, we need to define the specific heat, or the heat capacity, C_p in bits relative to the true temperature. Fortunately, heat capacity in bits has been developed by Fraundorf (Fraundorf 2000). Here is a summary. In a continuous system we would say,

$$C_v \equiv \frac{\Delta\Omega}{T} \text{ for no work or } C_v \triangleq \frac{\partial\Omega}{\partial T} \qquad (3.95)$$

$$d\Omega = \int_0^T C_v dT \qquad (3.96)$$

Since *T>0*, when $T \to 0, \Omega \to 0$, so

$$\xi = \frac{\Omega}{kT} = \frac{1}{kT} \int_0^T C_v dT \overset{T>0}{=} \frac{1}{k\Delta T} \int C_v dT \equiv \left\langle \frac{C_v}{k} \right\rangle \qquad (3.97)$$

where ξ is the heat capacity in bits over average temperatures ranging between *T* and absolute zero. *k* is unit preserving and relates the higher level messages to the fundamental primitive message measurement for the temperature. This means, we are able to relate heat capacity in bits to enable comparison of different measures of the conserved property; messages as records, messages as single primitive terms or messages as the true combinatorial set of sets of primitive single terms. This is adjusted using *k* in the temperature term $\beta=kT$. For the set of sets of true primitive terms *n*, *k=1*.

For this model, and likely for most systems whose internal structures can be represented with n-tuple q-levels of sets that can be developed from binary combinations, the appropriate granularity is sets of sets of message primitive *n*. The use of sets of sets of primitive messages also has the property of relating to the multiplicity of microstates which we say can be directly related to the entropy by taking the $\log_2$. Further, a

reasonably small alphabet, with a vocabulary consisting of as few as 32 terms gives a nice statistical sample set with a few messages.

This distribution function is also closely related to the more general Weibull probability distribution function.

$$P(q_i) = e^{-\left(\frac{q_i + \gamma}{\beta}\right)^{\alpha}} \qquad \text{a Weibull Distribution} \qquad (3.98)$$

We see $\beta = kT$ for the Temperature term in equation (6.1). Hence, we a have a relationship between Temperature from the microstates distribution at a given q-level.

We recall q-levels represent a set of q-level sets. A set composed from a pair of subsets is q-level =2, {AB}, a set made of three subsets {ABD} is q-level=3, etc, the more combinations, the greater Q_i, the more complex, the higher the temperature.

We can only go so far with the analogy to a physical system. Information and messages, are unlike a physical system. Our *software physics* based on information theoretic mechanisms has to differentiate over all of the various states of the q-levels. For example, {AB} is different and will occur with different probabilities in q-level=2. In a physical system, if two particles at energy state q=2, they are indistinguishable and all of the particles at that q-level have the same probability. This is what permits Boltzmann's equation $S = k \ln W$ to lose the summation operation seen in Shannon's famous equation. In the Shannon entropy equation, $S_H = -\sum_{x \in \Xi} p(x) \log_2 p(x)$, the coefficient when summing equal probabilities would end up being equal to one. The partitioning could just be across the energy levels. Although obvious now, this was a difficult problem to resolve to get the temperature to compute properly from both Shannon's view and the Boltzmann-Gibbs partition function view correctly.

In the proposed approach, while we bin all the sets of 2, and sets of three, etc, each set in the bin also has its own probability. If we did not do this, we would significantly underestimate the internal structural complexity, and hence the entropy of the system.

The temperature term, *kT,* turns out to be a constant for the system at a time step, which is

driven by α. This is visible in the appendix where the Weibull function is linearized to develop the curve fit.

We now have, in hand, a method to develop temperature in four ways.

1. From (3.25) where we look at the slope of the change of microstates to the change in the conserved property of two interacting subsystems.

$$\frac{1}{T} = \frac{\Delta S_H}{\Delta n} \tag{3.25}$$

2. The second approach looks at the dynamical system model. Here a pair of dynamical equations (3.80) represents the discrete interactions and seems to yield a relationship to temperature. We can partitioned down the macroscopic world to represent trajectories of microcanonical ensembles and their probability distributions at the node interaction level.

$$S_{k+1} = S_k + m_N N_k \tag{3.78}$$

$$N_{k+1} = N_k + m_S S_k \tag{3.79}$$

$$\frac{S_{k+1} - S_k}{N_{k+1} - N_k} = \frac{m_N N_k}{m_S S_k} = \frac{1}{\dfrac{m_S S_k}{m_N N_k}} = \frac{1}{T} = \frac{\Delta S}{\Delta N} \tag{3.80}$$

3. The third approach is through available occupancy microstates related to maximum entropy and the partition function, (3.88).

$$P(q_i) = Ce^{-q_i/kT} \quad \text{Boltzmann-Gibbs} \tag{3.87}$$

If we sum all of the q-levels, we get Ω_i given by

$$\Omega_c = \sum_{i \in \Omega_V} e^{-\frac{q_i}{kT}} \quad \text{Partition function} \tag{3.88}$$

and the more general distribution in the form of the Weibull function,

$$P(q_i) = e^{-\left(\frac{q_i+\gamma}{\beta}\right)^{\alpha}}$$ a Weibull Distribution   (3.98)

4. Closely related to all of these is the apparent relationship of temperature being proportional to pressure, where pressure is in terms of the conserved property per unit volume, or messages per node. This was seen in (3.34)

$$P(k) = \frac{m_P}{m_T}(T(k) - b_T) + b_P \qquad\qquad (3.34)$$

This is dimensionally correct as we saw from the partition function normalizing condition.

We also relate heat capacity in bits to enable comparison of different measures of the conserved property. This is adjusted using k in the temperature term $\beta = kT$. We see this in (3.97) $\xi = \frac{\Omega}{kT} \equiv \left\langle \frac{C_v}{k} \right\rangle$ which is the average heat capacity over temperature ranges from T to absolute zeros. This is valuable, since we can determine the heat capacity for a technology as we observe a sample over time. This then permits us to use the heat capacity to predict the number of nodes that must produce in order to get to our desired end state.

6.   Relationship of Marco and Micro through the bakers transformation

All of the pieces have now been developed. Let us bring it together using the bakers transformation. Equations (3.84) and (3.85) can also be written

$$\begin{bmatrix} S_{k+1} \\ N_{k+1} \end{bmatrix} = \begin{bmatrix} S_k\left(1 + \dfrac{m_s}{T}\right) \\ N_k\left(1 + Tm_N\right) \end{bmatrix}, \qquad\qquad 0 \le S_k < \left(\dfrac{1}{\left(1 + \dfrac{m_s}{T}\right)}\right) \qquad (3.99)$$

$$0 \le N_k < \left(1 + Tm_N\right)$$

$$\begin{bmatrix} S_{k+1} \\ N_{k+1} \end{bmatrix} = \begin{bmatrix} \left(S_k\left(1+\dfrac{m_s}{T}\right)-1\right) \\ \left(N_k\left(1+Tm_N\right)+\left(1+Tm_N\right)\right) \end{bmatrix}, \qquad \left(\dfrac{1}{\left(1+\dfrac{m_s}{T}\right)}\right) \le S_k \le 1$$

$$\left(1+Tm_N\right) \le N_k \le 1 \qquad\qquad (3.100)$$

This is of the form of the bakers' transformation. In this more general case, where p_s is a rational number

$$\begin{bmatrix} x_{k+1} \\ y_{k+1} \end{bmatrix} = \begin{bmatrix} p_s x_k \\ y_k / p_s \end{bmatrix}, \qquad 0 \le x_k < \dfrac{1}{p_s}$$

$$\begin{bmatrix} x_{k+1} \\ y_{k+1} \end{bmatrix} = \begin{bmatrix} p_s x_k - 1 \\ y_k / p_s + \dfrac{1}{p_s} \end{bmatrix}, \qquad \dfrac{1}{p_s} \le x_k \le 1 \qquad\qquad (3.101)$$

instead of

$$0 \le x_k < \left(\dfrac{1}{p_s}\right) \quad \text{we have} \quad 0 \le S_k < \left(\dfrac{1}{\left(1+\dfrac{m_s}{T}\right)}\right), \quad 0 \le N_k < \left(1+Tm_N\right) \qquad (3.102)$$

and

$$\left(\dfrac{1}{p_s}\right) \le x_k \le 1 \quad \text{we have} \quad \left(\dfrac{1}{\left(1+\dfrac{m_s}{T}\right)}\right) \le S_k \le 1, \ \left(1+Tm_N\right) \le N_k \le 1 \qquad (3.103)$$

.

Here $p_s = \left(1+\dfrac{m_s}{T}\right)$ and we can see the relationship of (3.78) or (3.84), (3.79) or (3.85), and (3.80) to reversible and irreversible, portation and production (3.86), mutual information and Baysian conditional probability, and chaos and order. The bakers transformation is related to a unit square with Euclidean distances. In our case, the control volume defines the unit square. We have the phase space representation of the

mapping of (3.101) locally expands horizontal segments by a factor of $1/p_s$ and contracts vertical (stable) ones by p_s. These are the chaos and order components respectively.

The bakers transformation is related to Bernoulli shifts (Prigogine 1989 p202). The simplest class of Kolmogorov systems (Elskins 1986) is Bernouli shifts. The relationship between the dynamical systems and information theoretic (Shannon 1948) and (Jaynes 1957) is known and directly exploited here for the foundation of technology transfer dynamics *TechTx* and the foundation for *software physics.*

It also makes sense. Now, we see that we are defining an evolutionary process. There appears to be a temperature, which we can represent in bits. We can define a specific heat for the entities under question. The process is really a program. The program takes information in and the length of the program and the entropy will be determined by the maximum entropy, the point where every state is known.

The idea that Kolmogorov has is there are objects and there are descriptions (encodings) of objects, and the complexity of an object is the minimal size of this description. If we have one publisher, and the publisher encodes a message, we can sum all of the publishers and messages (a countable number) and say some real things about the ensemble of messages (objects) and publishers (elemental control model nodes). This can be represented by a program (this process) in a finite length for the nodes and messages generated.

On an intuitive level, (per Uspensky) the elements of a "space can be taken as informations, and $y_k \leqslant y_{k+1}$ means that the information y_{k+1} is a refinement of the information y_k (and hence y_{k+1} is closer to some limit value to which both y_k and y_{k+1} serve as approximations." This even sounds like technology maturation.

In the appendix, we also see how the mean squared fluctuation of a property is related to the free "available" microstates. Future research experimentation can explore the actual values in the relationships for various technologies and evolving processes.

IV.   DATA ANALYSIS AND VALIDATION

Data has been collected on a sample of 50,744 raw messages for the seven technologies identified below. For purposes of exposition of the data to validate the model, the case of Ada is reviewed in detail. Java is summarized and plotted.

Technology	Messages (raw)	Final Messages	Terms	Instances	Confidence Interval ±	Years
Ada Experiment 1	6,023	3385			1.7%	22
Experiment 2 – N		4195	1460	17,347	0.76%	
n_singles		"		17,347	0.76%	
n		"	74,735	118,141	0.3%	
Java – N	6,307	4852	2421	26,309	0.6%	6
N		"		272,773	0.2%	
Abstract Data Types	567	567	364	1949	2.3%	
				8457	1.1%	
Rate Monotonic Analysis	223	223	342	1079	3.0%	
				6400	1.3%	
Software Cost Models	273	273	394	1134	3.0%	
				7131	1.2%	
Software Work Breakdown Structures	36	36	63	134	8.6%	
				567	4.2%	
Software Technology Transfer	257	257	222	1041	3.1%	
				6996	1.2%	

Table IV-1 Technologies Examined

Ada provided the basis for a number of experiments. In the Ada experiment 1, sample, there were 3,385 source records (messages), with 1,460 terms (the alphabet size) measured in 13,554 experimental data instances for the calculation of the actual entropy contribution of a message.   The model predictions are of entropy at the macro level, and the microstates of the terms arrangements is the basis of computing the sample distribution the error and confidence interval.   The result is the error and confidence intervals are VERY small when the sample size is in the thousands

The experiment indicated with "*N*", studied the effects of messages as complete records. In the Ada "*n_singles*" study, single primitive terms were considered the messages. In the last Ada study, "*n*", the combination of sets of terms in a record identifier were considered primitive messages.

Once it was established that *n* (the extensive variable) was the approach that best represented the intensive variable temperature, then all of the technologies were studied with distributions of sets of sets.

Recall that a positive Lyapunov exponent indicates chaos, not convergence. So we could have technologies which result in a Lyapunov exponent that is positive. For those cases, we need to know the initial data for a time step 0, with an accuracy of *N+k* places in order to determine a result with an accuracy of *N* digits after *k* iterations.

A. EXPERIMENT 1 (SENSITIVITY TO ANOMOLOUS DATA)

1.   *TechTx Basic Entropy* Macro Level Data and Analysis

For the validation of the *TechTx Basic Entropy* model, basic curve fitting is performed. The least squares method was used. Comparison of the sum of the residuals squared gives us a R^2 value to determine goodness of fit.

A discussion follows on the implications of the baseline model and the *TechTx Basic Entropy* model. The predictive strengths of each are presented. The other technology areas studied are then compared to the baseline model using the *TechTx Basic Entropy*.

2.   Data Source and Analysis Tools

All of the data came from the IEEE INSPEC database, for Physics, Electronics and Computing. It is a well-indexed database and has comprehensive coverage of the field. This is a database, which corresponds to the three print publications: *Physics Abstracts, Electrical and Electronic Abstracts,* and *Computer and Control Abstracts.* http://library.dialog.com/bluesheets/htmla/bl0004.html This family of science abstracts

began publication in 1898. There are approximately 4,100 journals and serials scanned, of which 750 are abstracted, cover to cover. This constitutes 82% of the database, including 6% from conference papers reported in journals. Another 16% come from conference proceedings. Books, reports, and dissertations are also covered.

The IEEE INSPEC database is an appropriate source for messages on a technology in the software-engineering field. Should a technology be desired outside of this area, another set of databases should be explored. INSPEC does use a controlled vocabulary from the INSPEC thesaurus. A single classification scheme is used for all records from 1969 to present. The IEEE INSPEC database updates are done 50 times per year. Each update averages about 6000 records.

The data was collected from the INSPEC database using the Naval Postgraduate School access to the Cambridge Scientific Abstract version of the INSPEC database from June through September 2000. To reproduce the data, any search engine that searches the INSPEC database should suffice. The raw data was processed by the US Army's open source intelligence engine TAOS. The TAOS system is available to the Army Tank Automotive Research, Development and Engineering Center. The TAOS version used is identical to Tech OASIS version 2.3a. Information on this system can be found on the VantagePoint web site at www.searchtech.com. For additional information on the TAOS system, see (Watts 2000, Porter 2000, Porter 2000a, Porter 200b, Porter 2000c). Contact the NextGenSoftware@TACOM.Army.mil and ask for the program manager. The point of contact for TAOS is Mr. Robert Watts.

This engine takes records, given from a simple Regular Expression application from a set of tagged, parses the data and identifies any field indicated in the regular expression schema. TAOS[23] was used to identify duplicate records, i.e. messages, in the context of this research.

B. ADA

---

[23]   <http://www.searchtech.com>. The current version of Tech OASIS is 2.3a. VantagePoint and TAOS, (Tech OASIS) are identical through the current version.

In experiment 1, 22 years of Ada data was drawn from the period of 1979 to 2001. This data set incorporated a deliberate wobble in terms of noisy data in order to reflect real world effects. The data for 2001 was left out of the analysis since it was not a complete year. The entropy data contains 32,076 data points based on 1458 terms, for 3385 non-duplicated records.

In experiment 2, over the same period, there is 34,862 data points based on 1583 terms, with 17,592 instances, in a total of 4249 non duplicated records. This resulted in 117,637 state points for the sample distribution. The comparison of the message – counting method and *TechTx Basic Entropy* model is done with the experiment 1 data set.

The equations in Chapter III can be easily adjusted to represent a portfolio of technologies; however, this is beyond the scope of this dissertation. The detailed data for Ada to compute the entropy is shown in the appendix. We need to understand that the curves we get for the data are local to the technology and vocabulary of the technology in question. We would not expect to see the same coefficients, exponents or intercepts for another technology.

1. Data and Method to Retrieve and Reduce Data

The data collected for Ada is typical of the method used for all of the other data sets and will be explained here.

In the case of Ada, only the term "Ada" was searched for anywhere in the database record. All of the records referenced were retrieved. There were 3385 unique records found in the database from 1969. Although the search was not refined by limiting the terms searched for, the first record that included the term "Ada" and dealt with "software" was in 1979. Prior to 1979, Ada referred to a number of different acronyms unrelated to the technology in question. A good search run by an information specialist or special librarian would throw out these "false drops". This stresses the importance of a good search strategy and identifying only relevant messages on the technology to be assessed.

The raw data was examined for duplicate records. These duplicates were removed. A Regular Expression application parsed the data, which was delimited by field tags and easily identifiable sub field delimiters and put into a flat file format. This format was readily examined by TAOS. The records were collected into time step bins. These bins were aggregated in annual, monthly and weekly time steps.

The first review of the annual data was done by publication year (PY). This is field assigned by INSPEC and entered by the indexers from the source document. Initial studies were all done using the publication year as the time step. From the confidence discussion in the next section, publication year time step bins are felt to be suitable for general-purpose use of the model approach described. For experimental purposes, more refined studies were done. During the later stage of experimentation, time step bins were identified from INSPEC accession number ranges. These ranges were determined from the INSPEC database by limiting the search. Here is an example search statement on the Dialog®, (www.dialog.com) information retrieval system.

s ud=199701w1
- *This gives you an update of the number of records added in "1997" during month "01" and week 1 "w1".*
- *To see the accession numbers, you can display the first and last accession number.*

d s1/1/Total =
- *This statement will give you the least recent (first one added) accession number in the set. "s1" is the set number, "1" is the code.*
- *To display the accession number and "Total" is the total number of records in the set.*

d s1/1/1 =
- *This statement will give you the most recent (last one added) accession number in the set. "s1" is the set number, "1" is the code to display the accession number and the second "1" is the first record in the set.*

This approach was performed to develop annual time step bins using accession number ranges. While it is clear that ranges of accession numbers for monthly and weekly time periods can be precisely identified, for a time step, it is a most time consuming process. Therefore, an approximation was made for monthly or weekly

intervals using accession (AN) number ranges. In each case, a the annual accession number ranges were divided into 50 and 12 equal parts for week and month time steps respectively.

### 2.	Interpretations of Data (Ada)

During experiment 1, the observed data had some wobble around 1985. This was the result of no records being captured from INSPEC for that year during our initial search. This wobble provided some interesting insights. The initial reaction was to throw the data out and start over. While we did collect more data, the wobble enabled visibility into the effects on the models caused by gaps in data. The wobble also seemed to represent the type of information that a practitioner would get as well, when not in the sterile conditions required by an experiment. While in a production system, we might want to "take what you get", for the purposes of sorting out the model and early usage, the data was closely examined. For each year without data, we averaged the data for the three years prior and two years after, as an estimate for the value of 1985. These adjustments have the same effect on all of the model studies, as you will see.

In experiment 2, pure data was collected to better refine the validation of the *TechTx Basic Entropy* model.

### 3.	Traditional Model – Message-Counting

Figure IV-1 illustrates the Ada data using the traditional method of Rogers (Rogers 1983, 1995), generally used for diffusion of innovations. This is also the method used by the researchers at Carnegie Mellon University, in Shaw's briefing on software architectures (Shaw 2001). The regression on the message count vs. time for Ada, using least squares fit, achieves an R^2 of .97. While this is usually considered a reasonable R^2 value, we shall see that the entropy approach also has a good fit and used together both provide predictive capability. Figure IV-2 shows the ability to project the future with 5 years and 10 years of data using the linear form fitted to the points marked with triangles.

For purposes of a cursory comparison of the two approaches, the entropy data is shown as circles and plotted against the secondary Y-axis. The secondary Y-axis scale was carefully chosen to have the two series' final years of data to nearly coincide. This permits a gloss-over discussion of the shape and influence of the data. This gloss seems to suggest that both models are subject to the same data anomalies.

A casual examination of the data seems to suggest an "S" type curve. It slowly starts, ramps up to nearly linear in the center section of the data and starts to tail off at the end. The tail off at the end could be explained by the fact that there is some lag in the publication, indexing, and database update process. For example, the last full year may not have all of the records posted from the prior year when the data was collected in June. Although 2001 is not in the data set, this most recent year that the data could have a PY date certainly could not have all of the final year data posted. Study of this lag could be made to better explain the *message-counting* shape more thoroughly. The same effects for process lag influence the entropy data. Both the message count and entropy data are influenced by the wobble in the data around 1985. In both cases, it tends to propagate into the future, since both models use cumulative information. Using the cumulative approach seems appropriate since the messages are persistent and available to all of the future researchers to examine.

One seemingly problematic area with these *message-counting* linear models is that the Y intercept is a negative number. This implies, that at time zero, there is a negative number of messages. While that is not possible, it could be suggesting there is some prior experience that will soon break loose. Prior learning in the entropy model seems to be more visible in that the entropy starts out driven by terms that existed prior to the subject technology introduction. This can be seen in Table IV-2. We know these were preexisting technology terms in Ada's heritage. A discussion of the fit for the *TechTx Basic Entropy* model follows the *message-counting* traditional model.

1980 Ada
1980 software-portability
1981 software-engineering
1981 programming
1981 military-computing
1981 standards
1981 operating-systems-computers
1981 multiprocessing-programs
1981 parallel-processing
1981 synchronisation
1981 computer-architecture
1981 multiprocessing-systems
1981 microprocessor-chips
1981 microcomputers
1981 military-equipment
1981 virtual-storage
1981 microprogramming

Table IV-2 Terms Identified in the Entropy Model for 1980, 81 (Years 2,3)

Traditional Method -- Count the Messages
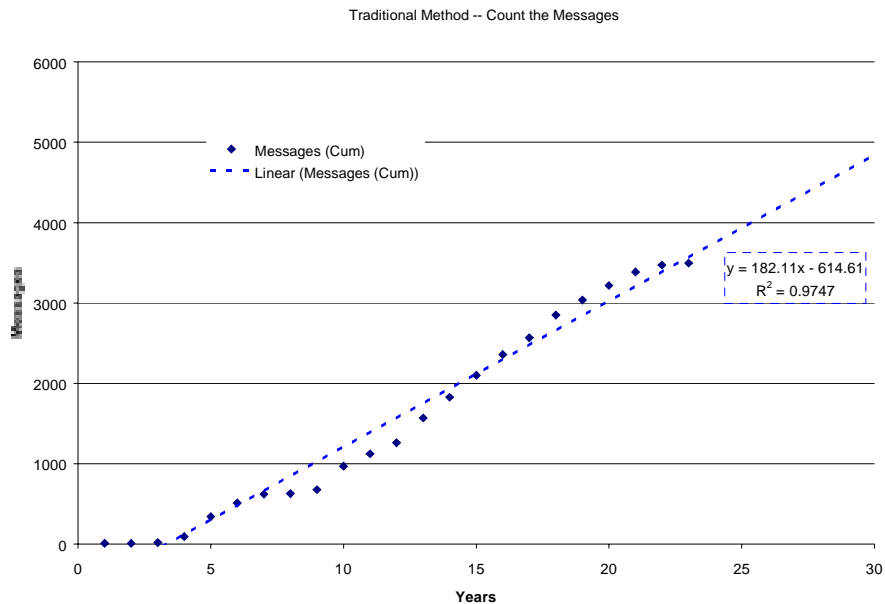
Figure IV-1 Traditional Model – Message-Counting
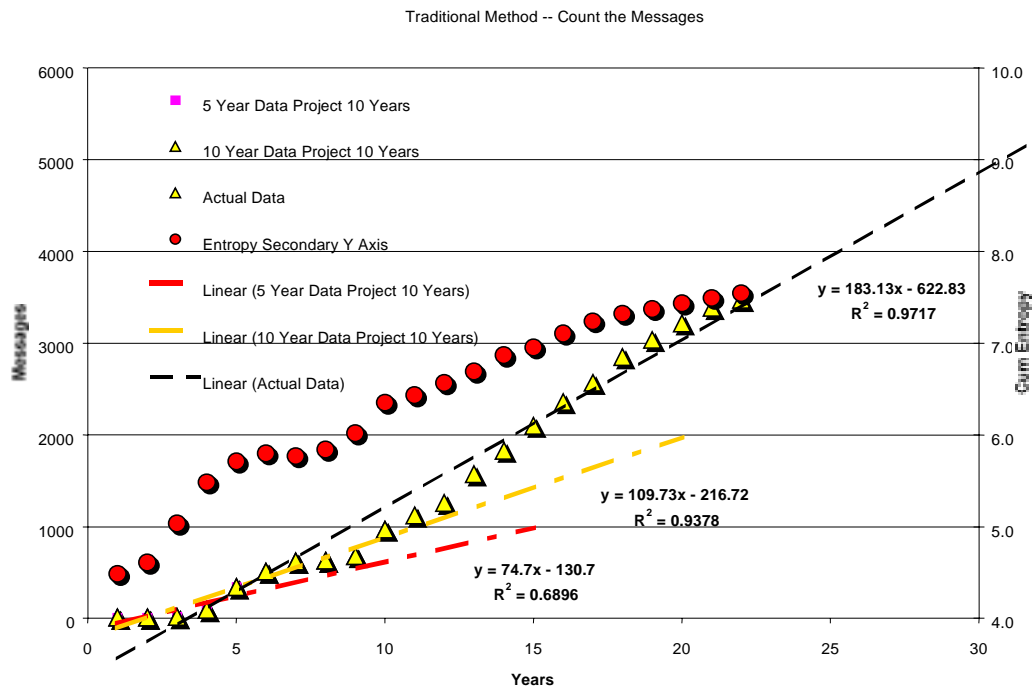
Figure IV-2 Traditional Model -- Projections using *message-counting* Approach

4. Improved TechTx Method – Basic Entropy Model

The entropy approach is driven by the terms contained in the messages. The data and trends in the distribution of the top 100 message terms for the Ada example is shown in Figure IV-3. This figure shows the cum entropy of the top 100 terms used in the messages distributed over time, with the start-time step of the data set at the back wall. The terms were sorted by their instance frequency. This loosely related to the information they contribute to the message pool over the period examined.

The terms (S1-S100) are sorted by the highest frequency of terms toward the left and lower frequency occurrences to the right. This is for the entire data set over the 22-year period. Spikes seen farther off to the right indicate early-use terms when the entire vocabulary was relatively lean. They quickly diminish in importance as time marches forward. It is interesting to look at the tabular entropy values from a 30,000 foot level. The Ada data is shown in the appendix to enable this view. The early terms seem to

show pedigree, e.g. Pascal. A late arriving term shows up with a lot of white space. If it is a melding (grafting) on another technology area that is rapidly growing, then the term arrives and stays in the higher frequency ranges, e.g. object orientation.

The first term is Ada, as would be expected, with a cum entropy contribution of .4276. This represents a gentle drop from a high entropy of .496. This decline is to be expected as more terms are added, and the search term influence is diluted. It would be surprising if the search term in question lost its position at the number one slot over the evaluation period. This would imply that another, or likely many other, terms are in ascendance relative to the search term.
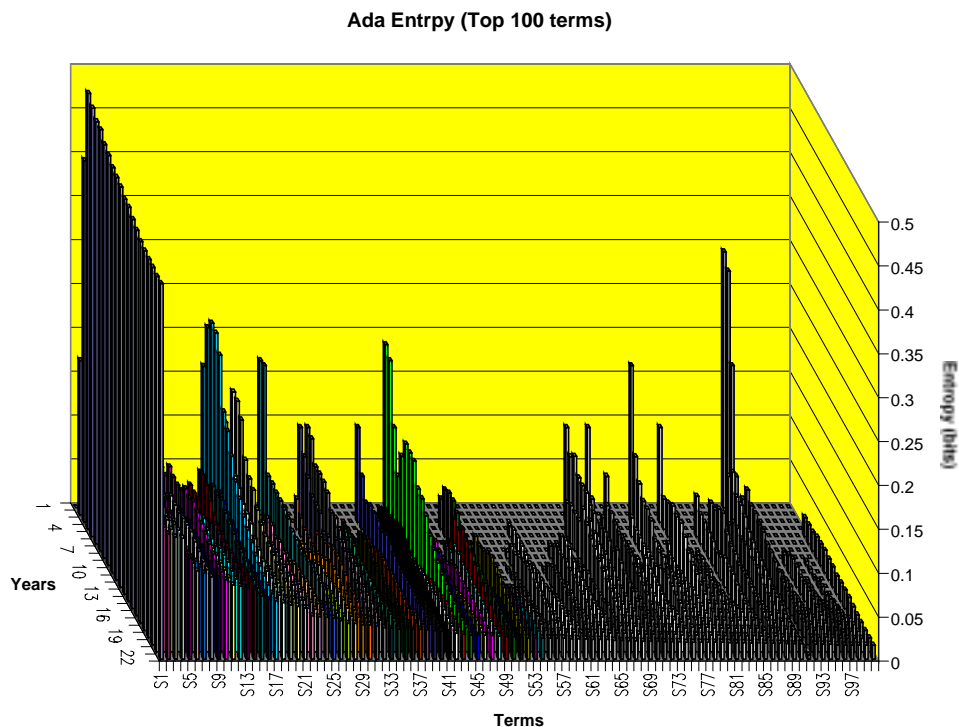
Figure IV-3 Top 100 Terms

A review of the top 50 terms also yields an expected result. The strengths of Ada are most often cited as seen in Table IV-3. Close examination of the data using the column labeled "slope" seems to provide insight into whether a related term is on the rise. This indicates that that term's contribution is adding to the technology, or in this case declining relative to Ada.

Order	Instances		Average	Slope	Influence	Max Entropy
1	2194	Ada	0.418	0.283	+	0.497
2	368	real-time-systems	0.089	0.093	+	0.148
3	355	software-engineering	0.195	0.088	+	0.304
4	351	object-oriented-programm	0.066	0.095	-	0.139
5	273	program-compilers	0.137	0.074	+	0.217
6	229	programming	0.119	0.066	+	0.176
7	221	object-oriented-languages	0.023	0.072	-	0.097
8	209	software-tools	0.059	0.060	+	0.104
9	204	aerospace-computing	0.054	0.060	+	0.098
10	199	military-computing	0.090	0.058	+	0.114
11	182	formal-specification	0.041	0.056	+	0.085
12	178	parallel-programming	0.048	0.055	+	0.086
13	174	software-reusability	0.045	0.054	+	0.088
14	168	computer-science-educati	0.062	0.055	-	0.104
15	148	programming-environment	0.055	0.046	+	0.099
16	137	distributed-processing	0.061	0.044	+	0.091
17	133	high-level-languages	0.121	0.042	+	0.251
18	128	data-structures	0.070	0.041	+	0.110
19	123	program-testing	0.038	0.041	+	0.062
20	118	digital-simulation	0.084	0.039	+	0.173
21	113	software	0.040	0.038	-	0.065
22	92	Ada-listings	0.035	0.032	+	0.070
23	89	C-language	0.021	0.033	-	0.048
24	88	program-verification	0.025	0.032	-	0.048
25	86	object-oriented	0.014	0.034	-	0.046
26	79	software-portability	0.060	0.029	+	0.178
27	72	object-oriented-methods	0.015	0.028	+	0.042
28	72	software-maintenance	0.018	0.027	+	0.044
29	69	software-reliability	0.023	0.026	+	0.045
30	68	fault-tolerant-computing	0.031	0.025	+	0.052
31	66	standards	0.048	0.024	+	0.108
32	62	operating-systems-compu	0.055	0.023	+	0.108
33	60	automatic-programming	0.031	0.023	+	0.045
34	60	educational-courses	0.022	0.024	-	0.041
35	59	abstract-data-types	0.010	0.023	+	0.034
36	58	multiprogramming	0.023	0.023	+	0.039
37	58	scheduling	0.021	0.021	+	0.039
38	56	safety-critical-software	0.006	0.024	-	0.032
39	55	multiprocessing-programs	0.038	0.021	-	0.108
40	54	inheritance	0.009	0.023	-	0.032
41	50	program-debugging	0.030	0.020	+	0.051
42	48	software-metrics	0.014	0.019	+	0.033
43	46	graphical-user-interfaces	0.012	0.019	+	0.030
44	45	Pascal	0.077	0.017	+	0.186
45	44	program-interpreters	0.023	0.018	+	0.045
46	44	concurrency-control	0.011	0.019	+	0.027
47	43	command-and-control-sys	0.020	0.018	+	0.035
48	42	knowledge-based-systems	0.018	0.016	+	0.039
49	41	expert-systems	0.026	0.016	+	0.058
50	41	systems-analysis	0.032	0.016	+	0.058

Table IV-3 Top 50 Terms Based on Cum Entropy

The slope is the comparison of two rates of change: the rate of change for the term compared to the rate of change of the technology, in this case Ada. The following is the equation for the "slope".

$$slope = \frac{d(Term_{ave})/dt}{d(Tech\_Term_{ave})/dt} = \frac{(LastYear - Average)_{Term}}{(LastYear - Average)_{Tech\_Term}} \tag{4.1}$$

The *LastYear* is the last full year of the data set. The *MaxEntropy* column is the peak value of the term's contribution to the overall entropy of the time step. The "Influence" column is determined by whether the last full year of the data is greater than the value at some arbitrary, but recent history value (*Entropy_{last year}* –*Entropy_{lastyear-4}*). In this case, that is four years prior. If the technology in question did fall off of the top slot, the terms that are driving the decent would be obvious from both the "Influence" column and the "slope". It would be a clear sign that relative to these ascendant terms, the study technology was declining. It might also suggest that to be rejuvenated, some of the facets represented in the ascendant technology should be evaluated to be grafted into the study technology. For example, if Java were maturing faster than Ada, which we can see is happening from the macro data in the next section, the common features (terms) of the technologies could be capitalized on. In fact, what has been observed in the case of Ada and Java, is that combining the technologies, gives the best of both worlds[24].

From this discussion, it is obvious that the *TechTx Basic Entropy* model provides significantly more insight than the *message-counting* model. Both are communication diffusion models, but the entropy approach provides more insight with only slightly more data parsing.

---

[24] This is based on discussions with Tucker Taft. Taft easily can be considered the chief architect of Ada 95. He has grafted Ada and Java at the byte code, virtual machine level. The result was a benefit to both languages.

Ada Entropy

$y = 4.1833x^{0.1854}$
$R^2 = 0.9669$

$y = 4.3478x^{0.1529}$
$R^2 = 0.9448$

$y = 4.3389x^{0.1567}$
$R^2 = 0.8922$

Entropy Bits

Year

● Power 5 Years Data Project 10 Years
● Power 10 Years Data Project 10 Years
● Entropy Ada Actuals
━━ Power 5 years Data Project 10 Years
━━ Power 10 Years Data Project 10 Years
╌╌╌ Power (Entropy Ada Actuals)
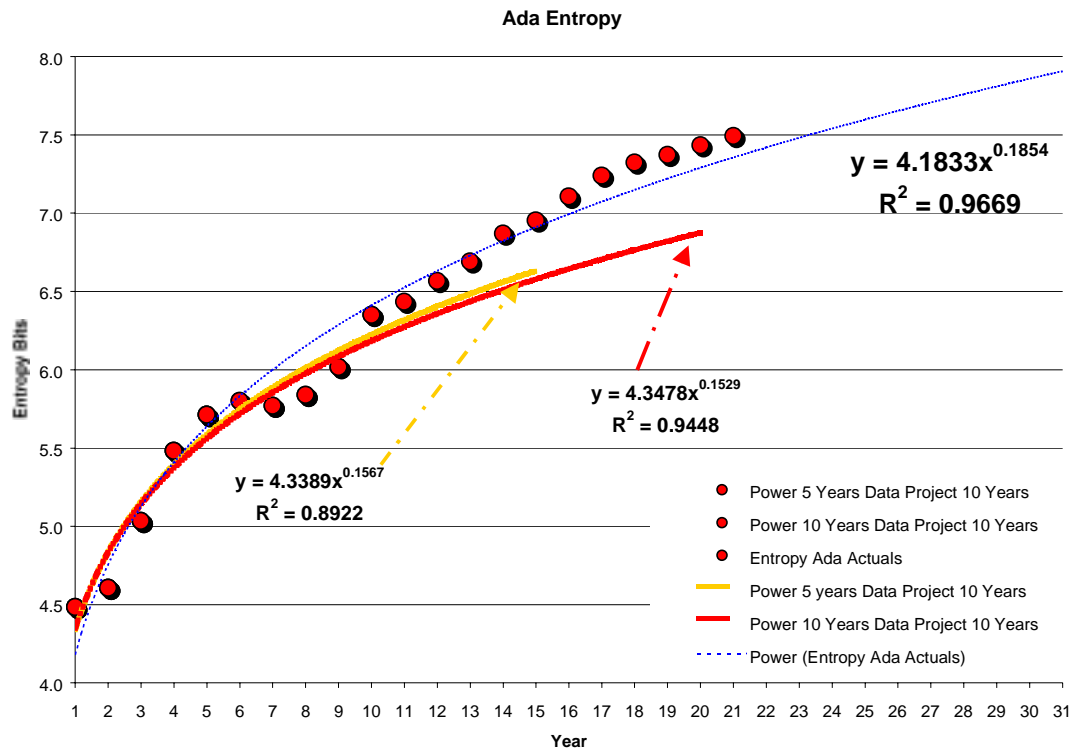
Figure IV-4 TechTx Basic Entropy Model Predictive Ability Experiment 1

Figure IV-4 illustrates the predictive capability of the basic entropy model. It is interesting to note that the entropy change (ΔS_H) vs Δ *time* is performing as one would expect. The rate of change is decreasing. This suggests stabilization. From this indicator, stabilization could mean two things. One is that the vocabulary and use of the technology has settled down. The other is that the pervasiveness once enjoyed in the early period is dissipated by other technologies. This has two effects. Ada, by definition, is affecting the other technologies, and they in turn are affecting Ada. This is an example of both the dissipative and integrative aspects of the bakers transformation discussed in Chapter II. Since we have knowledge about this technology (Ada), it is likely that both are occurring.

The curves for the comparison experiment 1, for Ada, can be seen in Figure IV-4.

Message Counting and Entropy Approach
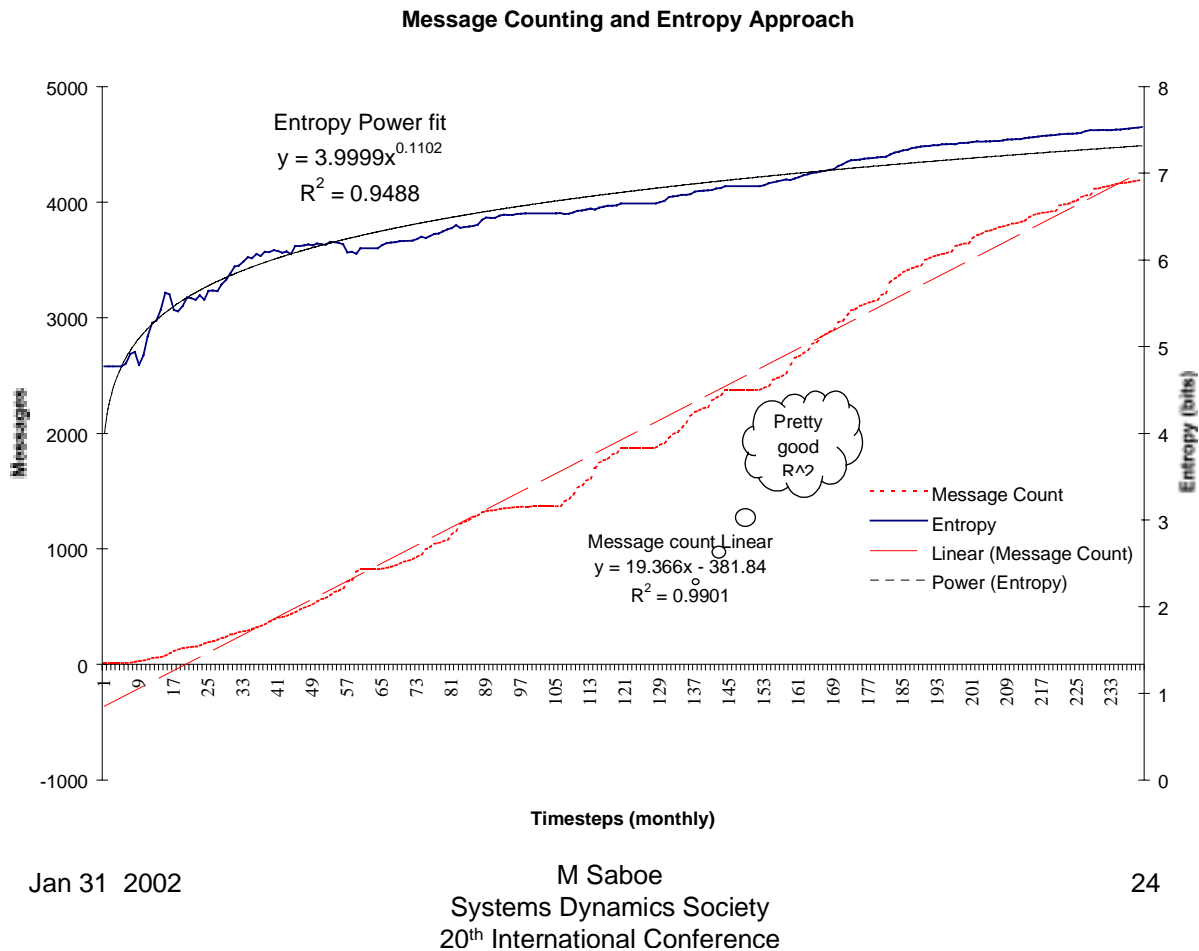
Figure IV-5 Entropy and messages *N* over time

One of the ways we determined that the fit for messages vs. time was linear follow. When we fit the data by taking all of the data and fitting the curves, then shifting on year (12 time steps in this case) and fitting the curve the data consistently showed that the R^2 extremely well correlated. For the message counting approach, we had an average R^2 of 0.985 for a linear function. For the entropy, a power curve fit yield, on average R^2 of 0.962. This is seen in Figure IV-5.

You will notice that there are several "flat" spots in Figure IV-5. This does not detract from the development of relationships of the various extensive and intensive variables. Occasionally, there are gaps in data. Regardless, the curve fit still is quite good. In the real world, there will also be gaps in data. Later, these data for the flat spots

are needed in order to develop difference, change, in the state properties per time step. Those data points are approximated by the curve fit. Formally, this is called regression imputation or conditional mean imputation approach. Using regression analysis, ordinary least squares, we modeled the missing data by predicting the missing data from data observed. This is consistent with methods for small data sets (Myrtveit 2001).

A summary of the analysis is given in Figure IV-6. While we tried to fit other curves to the data, these clearly came out superior in the data examined.

R-Squared Values for Ada, World

252 total steps, 21 years.

Year	Starting Step	Number of Publications		Entropy	
		R_squared	Equation	R_squared	Equation
1979	3	0.9867	y=18.928x + -50	0.8665	y=0.0105x + 5
1980	13	0.9901	y=19.366x + -38	0.925	y=0.0094x + 5
1981	25	0.9919	y=19.774x + -21	0.9726	y=0.0085x + 5
1982	37	0.9926	y=20.108x + -20	0.9852	y=0.008x + 5
1983	49	0.9926	y=20.377x + 18	0.9886	y=0.0079x + 6
1984	61	0.9919	y=20.587x + 40	0.9883	y=0.008x + 6
1985	73	0.9914	y=20.852x + 61	0.9879	y=0.0081x + 6
1986	85	0.9907	y=21.126x + 83	0.986	y=0.0079x + 6
1987	97	0.9898	y=21.429x + 10	0.9829	y=0.0079x + 6
1988	109	0.9899	y=21.903x + 12	0.9814	y=0.008x + 6
1989	121	0.9875	y=21.741x + 15	0.978	y=0.0082x + 6
1991	133	0.9839	y=21.71x + 18	0.9708	y=0.0081x + 6
1992	145	0.9791	y=21.329x + 21	0.9617	y=0.0079x + 6
1993	157	0.9708	y=20.983x + 23	0.9459	y=0.0077x + 6
1994	169	0.9665	y=19.624x + 27	0.9335	y=0.007x + 7
1995	181	0.9747	y=17.472x + 30	0.9521	y=0.0058x + 7
1996	193	0.9876	y=15.381x + 33	0.9762	y=0.0049x + 7
1997	205	0.9918	y=14.301x + 35	0.9901	y=0.0046x + 7
1998	217	0.988	y=13.755x + 37	0.9864	y=0.0048x + 7
1999	229	0.9646	y=13.742x + 38	0.9605	y=0.0045x + 7
2000	241	0.9891	y=7.2797x + 41	0.8915	y=0.0035x + 7
		0.985295238		0.962433333	

Figure IV-6 Curve fit for Messages and Entropy with various data subsets

For experiment 1, we get a power law curve fit for Entropy (S_H) using the complete data set, we get

$$S_H = 4.183t^{.185}$$ (4.2)

For the power-law curve fit for Entropy (S_H) using 5 years, we get

$$S_H = 4.34t^{.157} \tag{4.3}$$

For the power-law curve fit for Entropy (S_H) using 10 years, we get

$$S_H = 4.35t^{.153} \tag{4.4}$$

The *TechTx Basic Entropy* model error for all of the predictions is in a range (from -8% to +5%), when we realize that we are trying to predict the future. Note that the model tends to err on the conservative side, i.e. all of the out year predicted errors are negative. This conservative predication is due to the wobble in the data in the 1985-1986 range. This wobble reverberated in the out years and drove the out year predicted values down.

Other studies were conducted to determine whether other forms of the regression curves would fit better. The forms evaluated were linear, power, exponential, logarithmic, and time series. The logarithmic faired favorably with the power form for the entropy model, for long ranges of data, but poorly when trying to fit limited data points and predict the future. The time series and polynomial obviously can fit the data very precisely. The time series lacks predictive value beyond the number of periods in the moving average. The polynomial can very accurately match the actual data points, but predicts with 5 and 10 years of data very poorly.

The balance of this section for the eight technologies, we use the pure, experiment 2, data set. The Ada data and resulting curves for experiment 2 is shown in Figure IV-7.
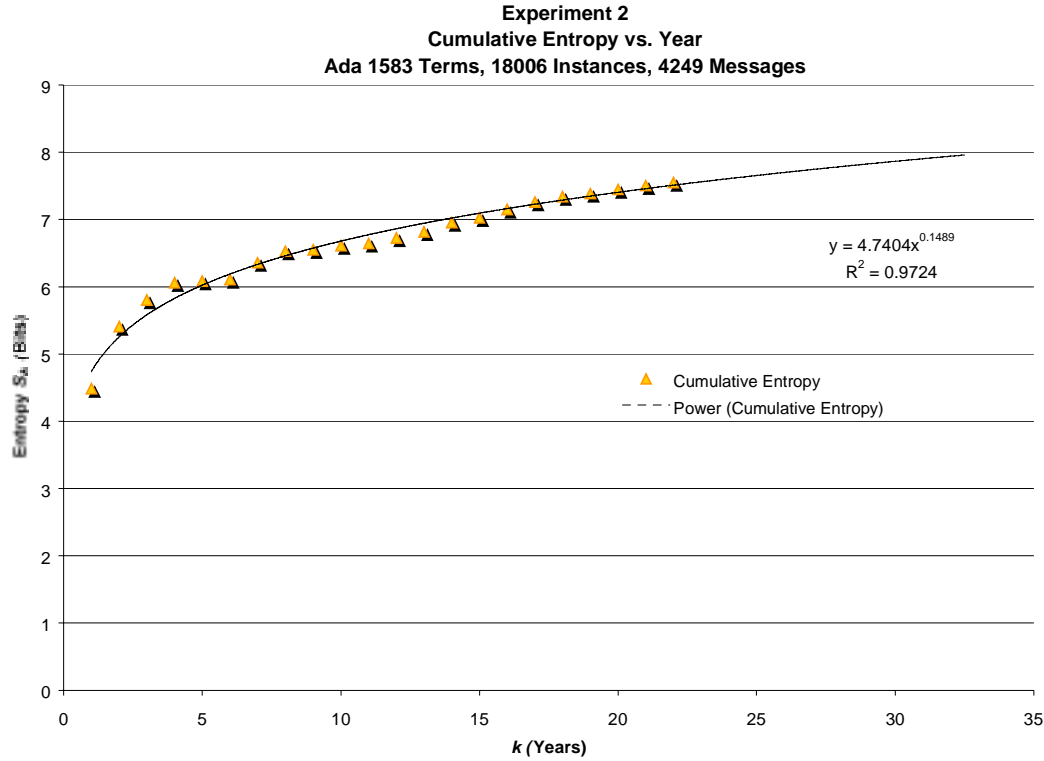
Figure IV-7 Ada *TechTx Basic Entropy* Experiment 2

5.   **Temperature from a Grand Canonical – Partition Function**

Let's look at temperature of the system and real data in yet another way. The idea of developing the maximum entropy can get a bit obscured with the empirical data, because of the fact that we are always adding terms and vocabulary. We shall evaluate the temperature-maximum entropy relationship of the distribution function by looking at the maximum entropy for a number of small alphabets first. Chapter III, indicated the number of potential microstates at each q-level for an alphabet of 128 terms and the maximum entropy for small alphabets of 128, 96, 64, 32, 16. We observed that the maximum entropy decreases as the alphabet increases. In a sense, this shows that as the alphabet size increases, the maximum q-level entropy decreases. So if terms are added to the vocabulary every time step, there is damping of the maximum entropy curve. The early (lower) q-levels will be filled much faster than the higher q-levels give smaller and smaller contribution to the entropy pool as the vocabulary increases as well. A way to think of this is filling a bath tub with hot, energetic molecules, but at the same time more

and more cool, low energetic, molecules are also being added. There comes a point where the hot particles are less of the population, and there is weighting to the lower q-levels.

Recognizing that in a system that has in influx of terms being discovered (we permitted them to exist in the alphabet when they were simply a potential concept set of terms), we will see a bias toward the lower q-levels. In a way, we may want to view this as a system that is in contact with a large reservoir, at ambient. Finally, the system is at an equilibrium with the reservoir, but since there are more states that are available in the technology system, it still attracts messages.

However, if we draw an appropriate control volume, around the terms in use, this indeed will approach maximum entropy. Simply start with a small number of terms, 4, 8, 16, 32, etc, and we can see that all of the states are soon occupied. Here we illustrate an alphabet of 4 terms, the top four terms in what will turn out to be well over 1000 terms in the technology's alphabet. However, for the purpose of validation and illustration Figure IV-8 meets the bill.

Max Entropy
Top 4 terms Measured
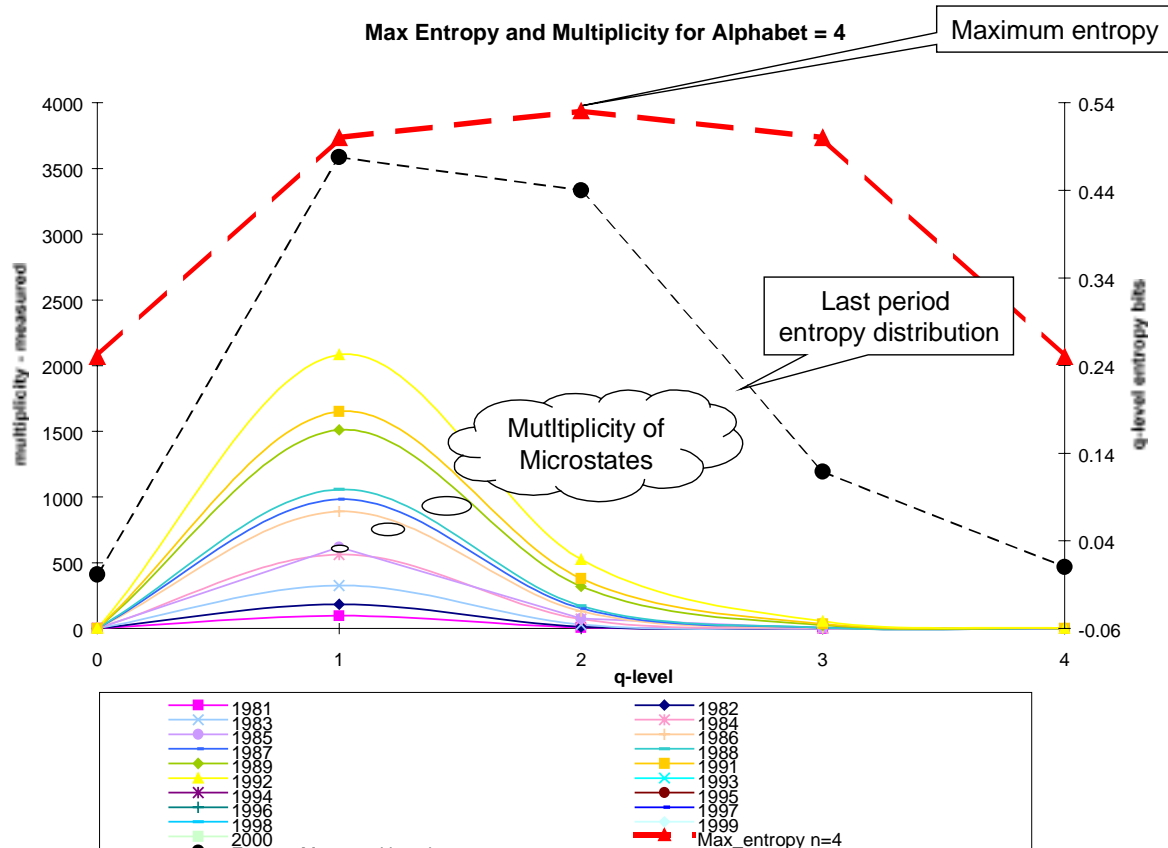
Figure IV-8 Max Entropy in a Small Alphabet (measured)

The upper curve indicated as a red dashed line with a ▲ marker represents the maximum entropy that an alphabet of four terms can have in a q-level. In the case of 4 terms, we get q-level =2 to have a maximum number of 6 microstates, out of a total multiplicity of 16 possible configurations. For q-levels 0 through 4, the microstates are 1,4,6,4,1.

The next black dashed line with ● marker represents the measured entropy for the most mature time step (yearly) in the technology's 4 term vocabulary. This is shown in Table IV-5. The first row is and indication of the q-level, where 0 is indicating a {}, null set. There is always one null set. The second row is the primitive message count of

subset combinations, with the sum at the far right. The third row is the entropy in a q-level. The last row indicates the q-level maximum entropy that can be achieved i.e. equi-probability of the sets.

q-level	0	1	2	3	4	sum
2000	1	4204	1352	166	8	7731
entropy_q-level	0.001671	0.4779	0.439922	0.118985	0.010261	1.048767
Max Entropy	0.25	0.5	0.530639	0.5	0.25	2.030639

Table IV-5 Measured Entropy, microstates, and maximum entropy

This shows us that the entropy does increases and starts to approach the maximum. Figure IV-9 shows how the temperature term computed from this small alphabet increases over time steps.
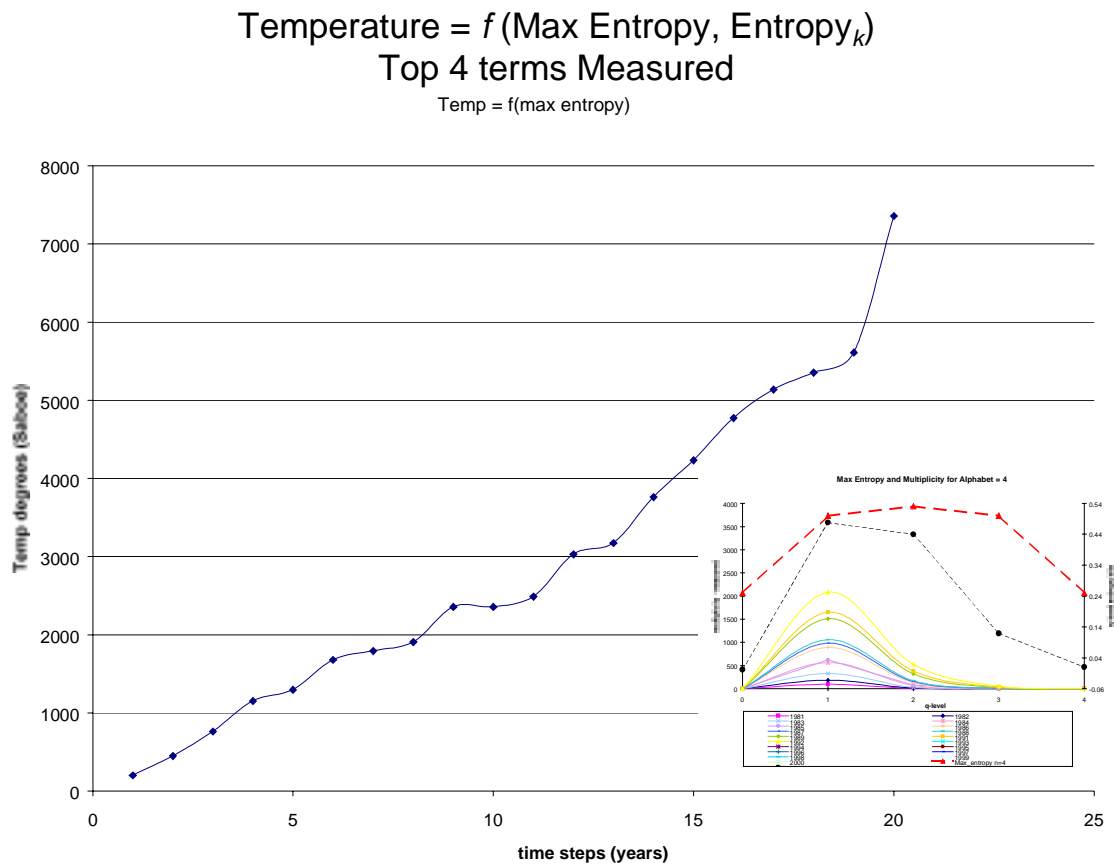
Figure IV-9 Small Alphabet (4 terms) from Ada, Temperature term vs time

6. Validating the Partition Function

We validate the partition function for the model using Ada. The microstates of the alphabet and vocabulary for Ada are related to the primitive terms *n* in various q-levels. A technology example of microstates, q-levels and entropy is shown in Figure IV-10. The number of microstates or the set entities (primitive messages) in the various q-level (*x*-axis) are shown on the left-hand *y*-axis. The cumulative entropy, that is computing the entropy of each q-level, is shown on the right *y*-axis.

Ada ntuple sets
Entities, q levels, Entropy (Cummulative)
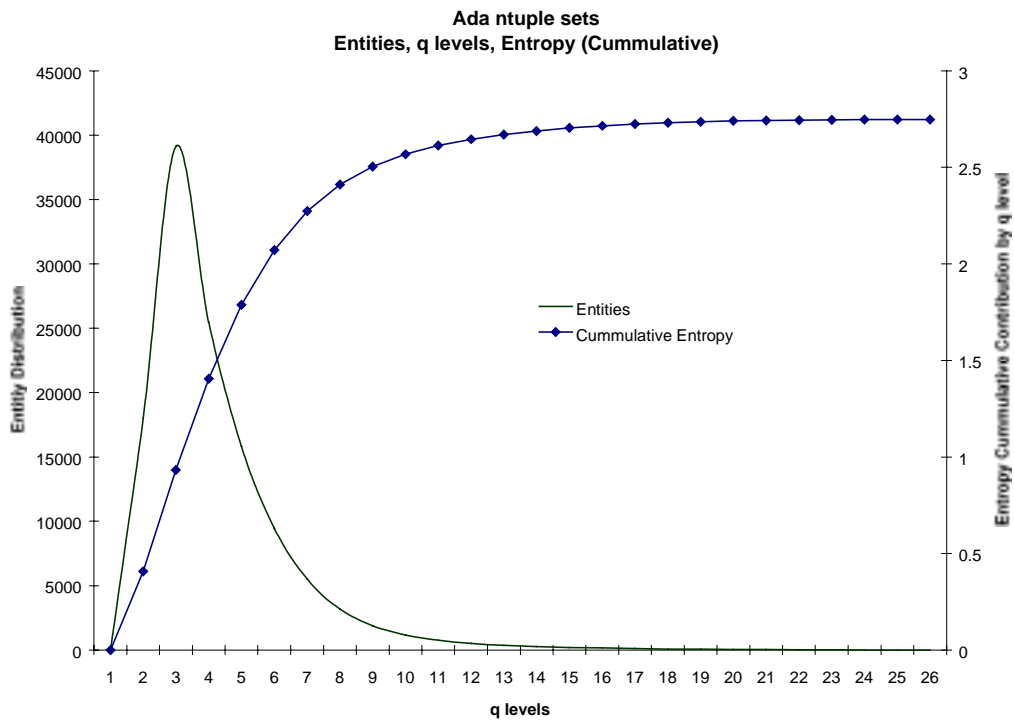
Figure IV-10 *n* microstates distribution to q-levels, and Cumulative Entropy

q-level Distribution

$\Omega_c =$ microstates at this time step $\quad \Omega_c = \sum\limits_{i \in \Omega_{V,N}} e^{-\dfrac{q_i}{kT}}$

$P(q_i) = Ce^{-q_i/kT}$ Boltzmann-Gibbs

$= e^{-\left(\frac{q_i + \gamma}{\beta}\right)^{\alpha}}$ a Weibull Curve

$\alpha = 1.01068 \approx 1$ $R^2 = .999397$

$kT = 108,313$ for this example

- P(q_l)_calculated
- P(q_l)_actual
- Probability(q_l)
- Cummulative Probability

left axis: microstates (q message primitives) — 0, 20000, 40000, 60000, 80000, 100000

right axis: Probability (q_i) — 0, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 1

x axis: q-level — 1.0, 2.0, 3.0, 4.0, 5.0, 6.0, 7.0, 8.0, 9.0, 10.0, 11.0, 12.0, 13.0, 14.0, 15.0, 16.0, 17.0, 18.0, 19.0, 20.0, 21.0, 22.0, 23.0, 24.0, 25.0

q_Level_fit_Actual_B.xls

Feb 2002   M Saboe   74
Ph.D. Defense 2002

Figure IV-11 q-level distribution, actual and modeled, probability and the Weibull distribution.

We validate the computation of a temperature using the partition function using Ada. In Figure IV-11, on the left-hand *y*-axis, we show the number of microstates populated by sets of terms. Each pair of bars represents a calculated microstates occupancy and actual q-level primitive message occupancy count. The calculated value is the bar the left of the pair. Tracing the bars is a probability of being found in the q-level. The probability is associated with the secondary *y*-axis on the right. We also can see the cumulative probability distribution function, which is the upper curve. The curve we can observe approaches 1, with each q-level having a smaller and smaller probability of being entered. The curve shows that there is over a 75% chance of being in q-levels 1 through 4. This distribution can be modeled as a Botzmann-Gibbs distribution function using (3.98). For the sample in Figure IV-11, we have

$$\alpha = 1.01068 \approx 1$$
. $$\gamma = 0$$
$$kT = 108,313$$

We end up with a $R^2 = .999397$, a pretty good correlation. Refinements to the model will enable determination of the temperature in *^{o}degrees Saboe*.

Temperature Sensitivity to Granularity
Temperature vs Entropy for Technology (Ada)
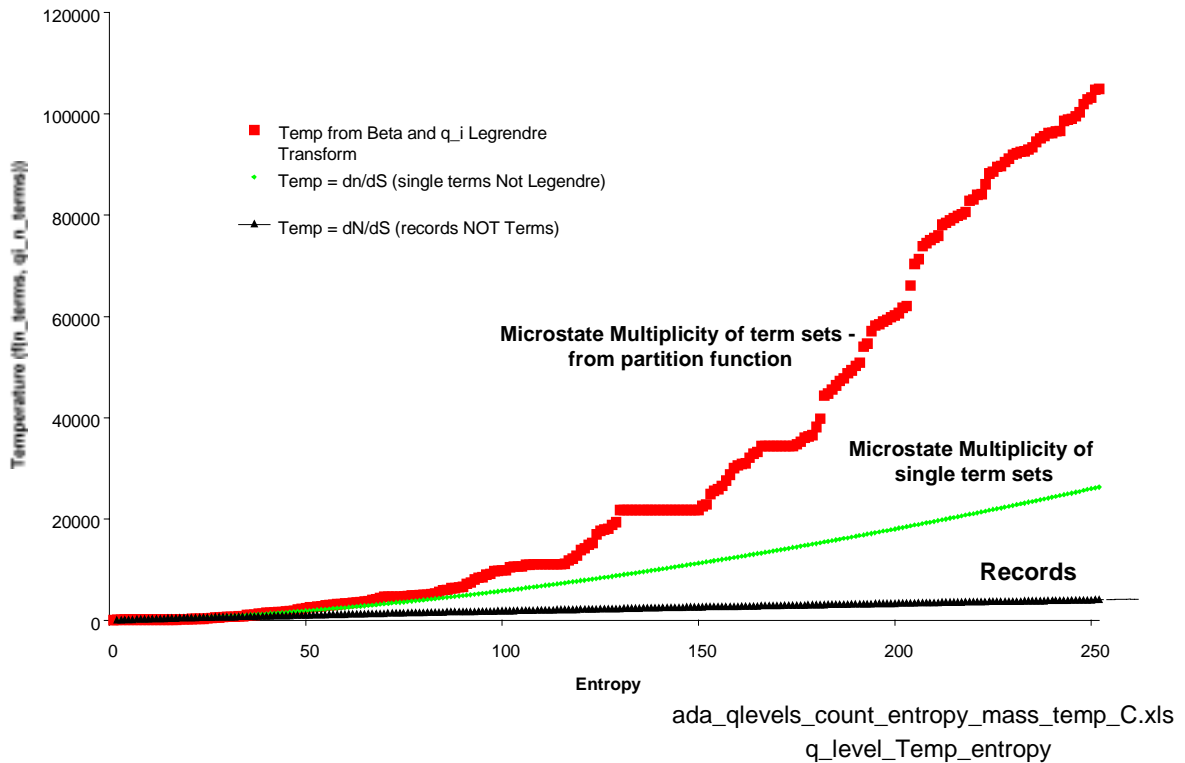
Figure IV-12 Temperature Sensitivity to Granularity.

Figure IV-12 shows the sensitivity of the temperature term to abstraction. Chapter III discussed the approach to counting the conserved extensive variable. The lower blue curve computes the temperature as a function for the trend line approximation of mean records N being processed. Similarly, the middle, green curve, relates to single term distribution of the trend of primitive messages. This is a finer granularity than a record, but does not yet meet the desired set of complete conditions to describe temperature. The upper curve, red curve, illustrates the actual data points, not the mean

of the trend line. This is the temperature term of the data when, the sets of sets of terms which we observed were allocated to q-levels. Even these are yet an approximation since all of the terms in the bin were given equal probabilities.

We also note that the exponent α also is greater than one. This is as suggested by Prigogine for a social system. We might restate Prigogine's comment more generally as "for a non-physical system the exponent α on the conserved property interactions might be greater than one."

Figure IV-13 shows that the data using the partition function does in fact perform as the theory in Chapter III predicted. As time passes, the technology heats up and consumes free energy states and also heats up due to the addition, across the control boundary of additional terms as answers !!! that previously were questions ???. We can observe that with time the trend is obvious and very predictable. The confidence level on this data is on the order of $1/\sqrt{118,141} = \pm 0.3\%$. Due to the construction of the model, the data will always have a very tight confidence limit. Even as few as 1041 primitive terms will yield $1/\sqrt{1041} = \pm 3.1\%$.

Ada Distribution of Messages by q-level

q level Distribution
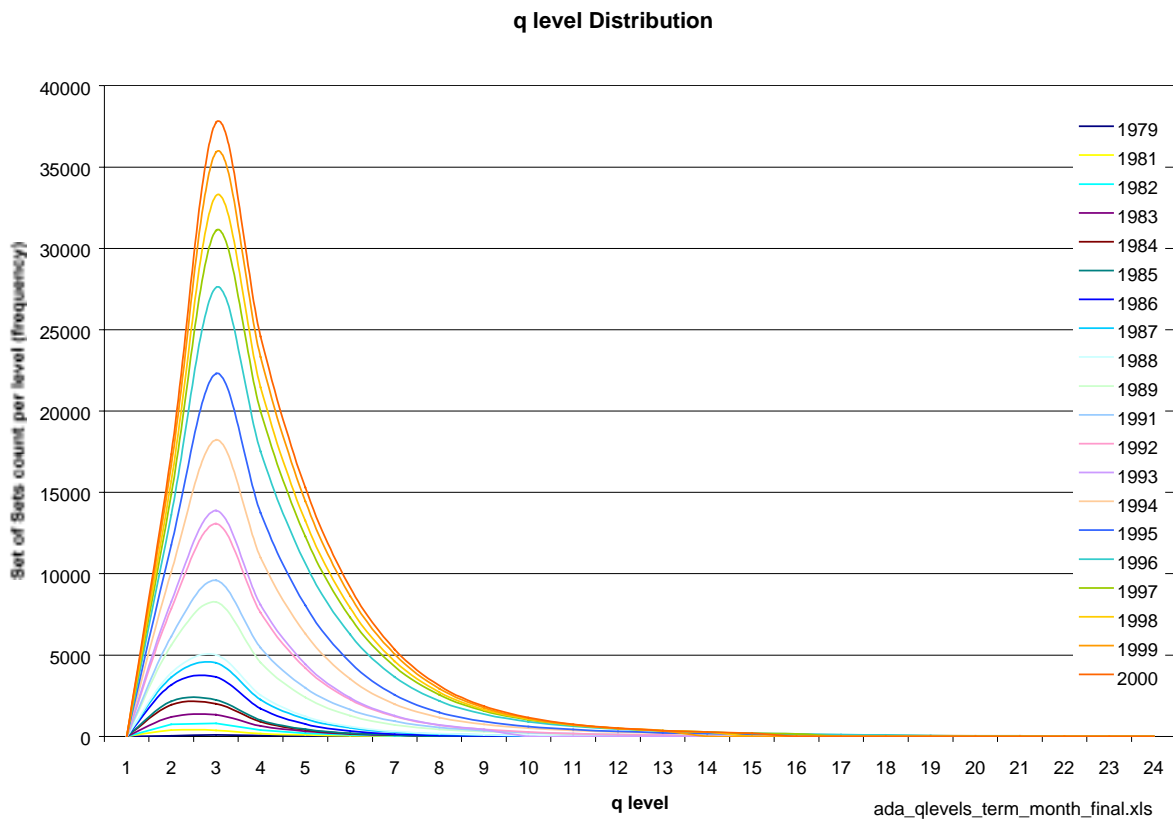
Figure IV-13 Ada Partition function validation

The following list of technologies were evaluated. The entropy and linear model curves are compared in Figure IV-14. A thumbnail for each technology is shown below.

Experiment 2
Cumulative Entropy vs. Year
Java 2813 Terms, 28907 Instances, 5330 Messages, 6 Years

$y = 6.6188x^{0.159}$
$R^2 = 0.9405$

$y = 4.7404x^{0.1489}$
$R^2 = 0.9724$

△ Java Cumulative Entropy
● Ada Cumulative Entropy
– – Power (Java Cumulative Entropy)
– – Ada 22 Years Data Project 10 Years

Entropy S_k (Bits)
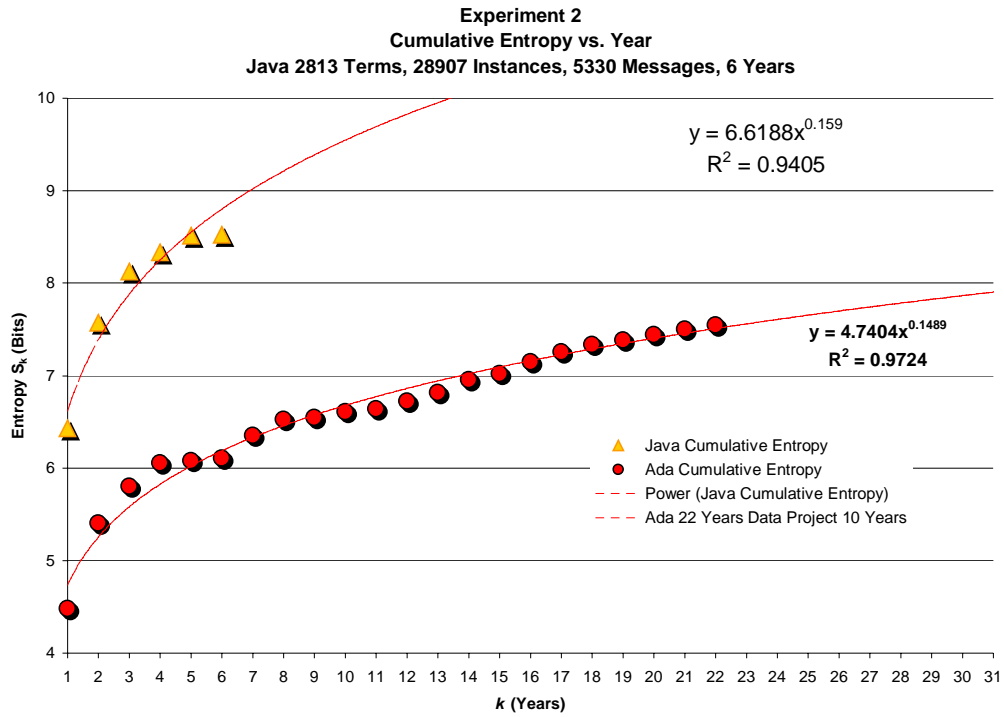
k (Years)

Figure IV-14 Java and Ada Comparison Entropy S_k *vs k* (time step = years)

- Java

Java relationships
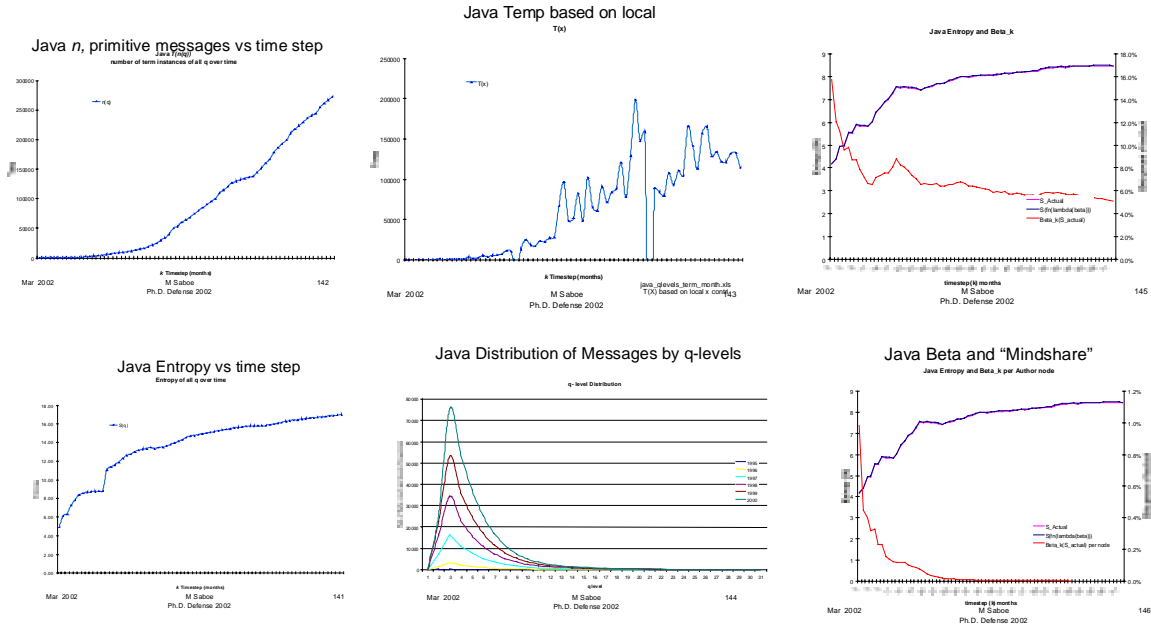
Figure IV-15 Java Relationships

For an early Ada example seen in Figure IV-16, we can observe that both curves, the curve for the Lyapunov exponent and Shannon's entropy have the same power law form. By observation, we see both of the entropy measures as a function of time step. S_H, the information theory entropy measure is on the left *y-axis*, and the S_B which comes from the eigenvalue of the micro control model (hence in the range of 0 to 1), is on the right hand *y*-axis. The scales were adjusted to easily see that both curves are of the same form. In addition, we can see for this early data set, that the R^2 values are reasonable, at 0.968 for system level entropy and 0.96 for the bakers transformation *j*. We can see that as the system entropy stabilizes, the eigenvalue of the feedback control dynamical system is also stabilizing.

Initially, to determine the *form* of the functions, the average value $\beta \approx 10\%$ was used. This was done by iterative guesses of a fixed β. This approximation of β was used

to satisfy the macroscopic rate of change of entropy. This suggests that we have the right form of the dynamical system control model matched to the macroscopic system model. This also suggests that the model does approximate the observed conditions.

Entropy S_H and Dynamical System j
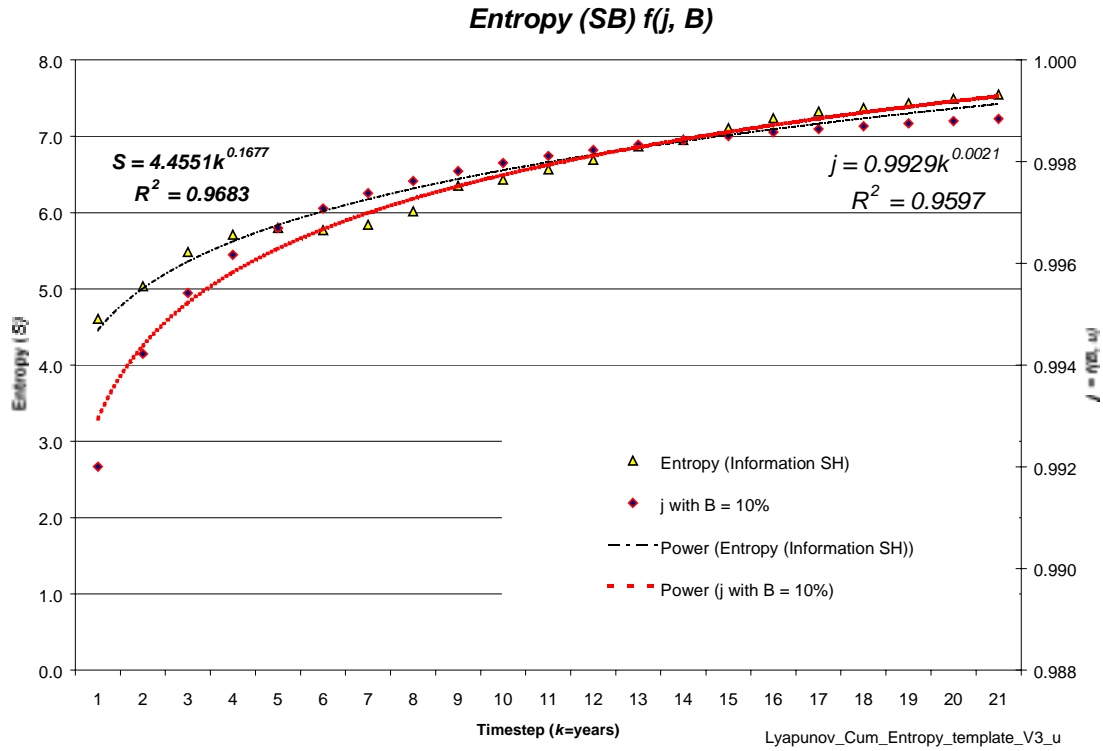
Figure IV-16 Macro Equilibrium S_H and Eigenvalue j Stabilization

From (3.77), which develops Shannon entropy now in terms of j_k, which we know from (3.70) is a function of β. At this point β was adjusted until the entropy (eigenvalue) of the discrete model matched the macroscopic entropy of the information theoretic model. In each time step, the tolerance on the two methods of computing the entropy were matched to within 0.1%. This is seen in Figure IV-17. The upper curves (the two are superimposed) represent entropy converging at the same timestamps for the system. The lower curve represents β, which changes over time. The secondary y-axis, on the right gives β as a percentage.

Entropy (Lyapunov, (Beta))

Entropy and Beta_k
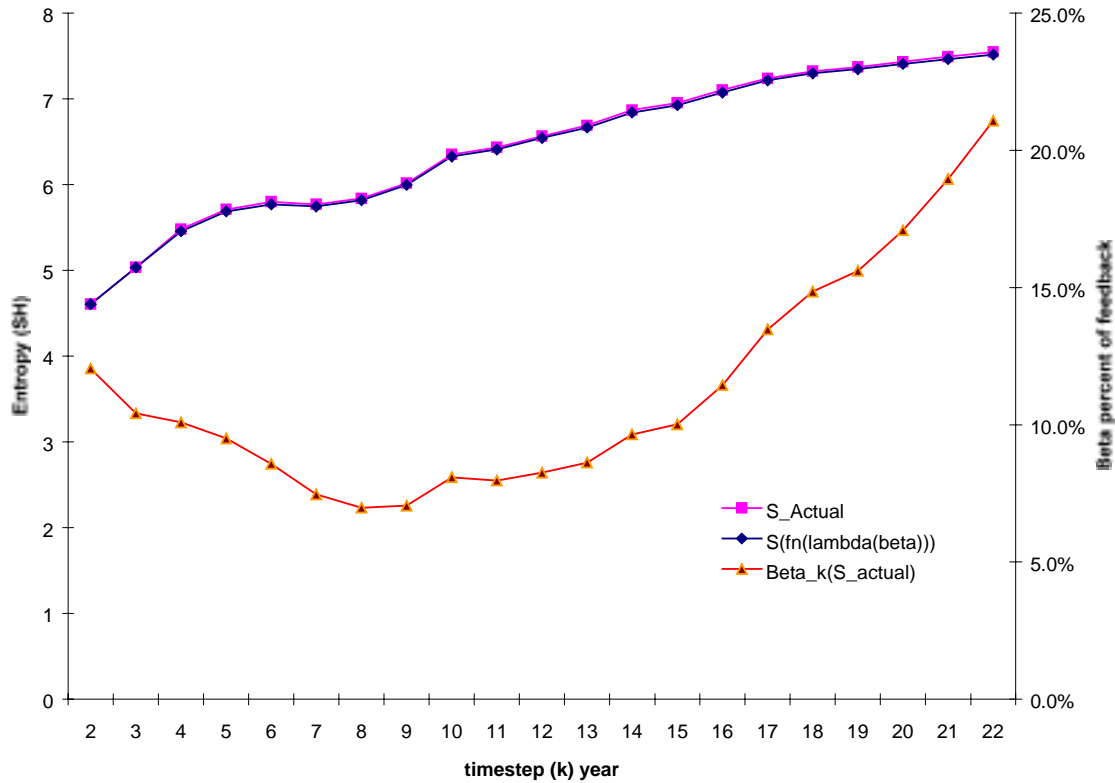
Figure IV-17 Solving for β to converge S_B and S_H

At this point, we are considering the "community" a large node. In the real world, the community is partitioned into a volume of performing nodes, and these nodes have different performance rates. However, at the community node level, we can not distinguish what the contribution is for mind share or learning. It would be useful to tease apart the contribution that is due to mind share and that which is due to learning.

A quick look can be obtained by allocating each author's contribution. This is done by dividing β by the number of authors and determining the $< \beta\ z_k >$, average feedback messages request. This result is shown in Figure IV-18. The dashed, red, curve

represents the allocation of β to each author, using the left *y*-axis. The right hand *y*-axis, gives the accumulation of the number of authors, or "mind share", which increases over time.

Beta and "Mindshare"

Beta per Author and Mindshare and

Ph.D. Defense 2002
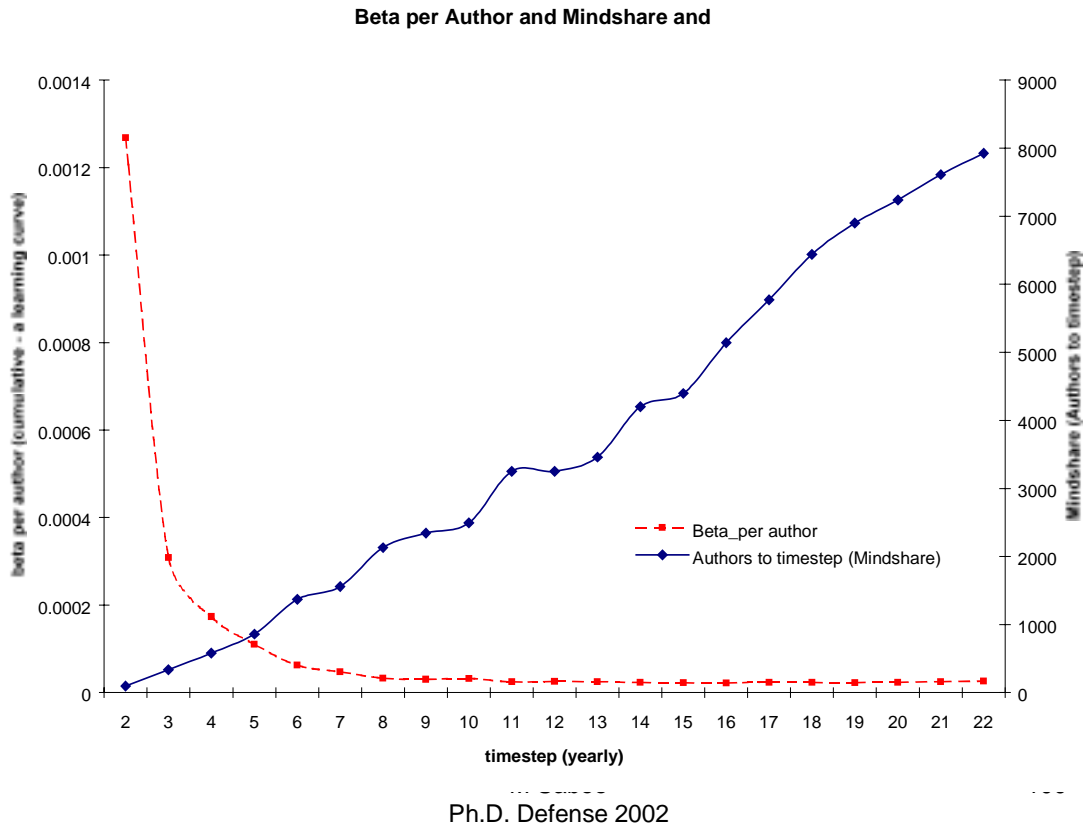
Figure IV-18 β Feedback requested from persistent messages, allocated per author.

We can see that βz_k is decreasing over time. β is decreasing with the number of total messages, or tasks performed. Learning appears to be occurring, or the messages are more easily understood. Understanding the message and immediately being able to act on it can be considered the result of learning, or improved packaging of the message. Discussing the various learning curves, is beyond the scope of this research. However, in Chapter VI, future research directions are suggested that may relate a form of the learning curve to entropy.

V.   SUMMARY OF CONTRIBUTIONS

The ability to bridge these two previously disconnected views of a physical and non- physical world conveniently provides powerful analytical tools to the software engineer. This is a nontrivial contribution to the software engineering community; we can put methods in the hands of software engineers that can be readily grasped by the mechanical, electrical, or communication engineer or anyone who has had some basic physics. This reduces the barriers to use by lowering the effort required to unpack, decipher and understand the "communication protocol" for the user community for this technology. In this technology case, the experiment was technology transfer. Since we used a communication by a set of symbols that were canonically related, and a method that is already common to the engineer and scientist, we have increased the available, high q-level microstates, which contain powerful concepts.

This research tied the three main components together in the *TechTx Entropy Feedback* model. These are information theory, statistical mechanics, with the dynamical control model of the technology transfer model. In a relatively comfortable way, we have tied in Rogers *Innovation* (the software information base element), his *communication* network of exchanges of information reducing uncertainty and improving the mutual information of the sender, receiver and consumer. We also address the *time* aspect. Recall the baker transformation iterations of folding, stretching, rotating and translating represented a mathematicians view of time.   In order to address *time* and all of the other observed aspects of technology evolution, we use the information theory entropy and the chaos control model.   This had a critical aspect that related the two views of entropy, which took time out of the picture in terms of clock time and related *time* to mixing and the bakers transformation.

The major contribution is the development of a series of equations of state that define evolutionary models. The key element was the approach to provide an engineer with a relationship of temperature, entropy and a conserved property.   Temperature is

fundamental information units and referred to as °*Degrees Saboe.* Temperature is significant because it relates the maximum complexity of a system to the current complexity. This is a proven metric that can be applied in many places to software engineering, e.g. software complexity. A direct relationship can be easily made to Halstead's metric which is familiar to software engineers. This in turn has been related to the rate humans are capable of making decisions between two choice, e.g. alphabet sets of sets of operator and operands, operators and edges, operators and flows.

These equations are enable the development of temperature of a process in four ways.

1. From (3.25) where we look at the slope of the change of microstates to the change in the conserved property of two interacting subsystems.

$$\frac{1}{T} = \frac{\Delta S_H}{\Delta n} \qquad (3.25)$$

2. The second approach looks at the dynamical system model. Here a pair of dynamical equations (3.80), represents the discrete interactions and provided a relationship to temperature. We can partitioned down the macroscopic world to represent trajectories of microcanonical ensembles and their probability distributions at the node interaction level.

$$S_{k+1} = S_k + m_N N_k \qquad (3.78)$$

$$N_{k+1} = N_k + m_S S_k \qquad (3.79)$$

$$\frac{S_{k+1} - S_k}{N_{k+1} - N_k} = \frac{m_N N_k}{m_S S_k} = \frac{1}{\dfrac{m_S S_k}{m_N N_k}} = \frac{1}{T} = \frac{\Delta S}{\Delta N} \qquad (3.80)$$

3. The third approach is through available occupancy microstates related to maximum entropy and the partition function, (3.88).

$$P(q_i) = C e^{-q_i/kT} \qquad \text{Boltzmann-Gibbs} \qquad (3.87)$$

If we sum all of the q-levels, we get Ω_i given by

$$\Omega_c = \sum_{i \in \Omega_V} e^{-\frac{q_i}{kT}} \qquad \text{Partition function} \qquad (3.88)$$

and the more general distribution in the form of the Weibull function,

$$P(q_i) = e^{-\left(\frac{q_i+\gamma}{\beta}\right)^\alpha} \qquad \text{a Weibull Distribution} \qquad (3.98)$$

4. Closely related to all of these is the apparent relationship of temperature being proportional to pressure, where pressure is in terms of the conserved property per unit volume, or messages per node. This was seen in (3.34)

$$P(k) = \frac{m_P}{m_T}(T(k) - b_T) + b_P \qquad (3.34)$$

This is dimensionally correct as we saw from the partition function normalizing condition. It also turns out that learning has the same dimensions. The time to perform a task is related to cumulative messages performed per node per time step.

The most significant contribution is the relationship of the dynamical systems model to the bakers transformation.

$$\begin{bmatrix} S_{k+1} \\ N_{k+1} \end{bmatrix} = \begin{bmatrix} S_k\left(1+\frac{m_s}{T}\right) \\ N_k\left(1+Tm_N\right) \end{bmatrix}, \qquad 0 \le S_k < \left(\frac{1}{\left(1+\frac{m_s}{T}\right)}\right) \qquad (3.99)$$
$$0 \le N_k < \left(1+Tm_N\right)$$

$$\begin{bmatrix} S_{k+1} \\ N_{k+1} \end{bmatrix} = \begin{bmatrix} \left(S_k\left(1+\frac{m_s}{T}\right)-1\right) \\ \left(N_k\left(1+Tm_N\right)+\left(1+Tm_N\right)\right) \end{bmatrix}, \qquad \left(\frac{1}{\left(1+\frac{m_s}{T}\right)}\right) \le S_k \le 1 \qquad (3.100)$$
$$\left(1+Tm_N\right) \le N_k \le 1$$

This is of the form of the bakers' transformation. In this more general case, where p_s is a rational number

$$\begin{bmatrix} x_{k+1} \\ y_{k+1} \end{bmatrix} = \begin{bmatrix} p_s x_k \\ y_k / p_s \end{bmatrix}, \qquad 0 \le x_k < \frac{1}{p_s}$$

$$\begin{bmatrix} x_{k+1} \\ y_{k+1} \end{bmatrix} = \begin{bmatrix} p_s x_k - 1 \\ y_k / p_s + \dfrac{1}{p_s} \end{bmatrix}, \qquad \frac{1}{p_s} \le x_k \le 1 \tag{3.101}$$

instead of

$$0 \le x_k < \left(\frac{1}{p_s} \right) \text{ we have } 0 \le S_k < \left(\frac{1}{\left(1 + \dfrac{m_s}{T} \right)} \right), \; 0 \le N_k < (1 + Tm_N) \tag{3.102}$$

and

$$\left(\frac{1}{p_s} \right) \le x_k \le 1 \text{ we have } \left(\frac{1}{\left(1 + \dfrac{m_s}{T} \right)} \right) \le S_k \le 1, \; (1 + Tm_N) \le N_k \le 1 \tag{3.103}$$

.

Here $p_s = \left(1 + \dfrac{m_s}{T} \right)$. We saw the relationship to reversible and irreversible entropy components, portation and production, mutual information and Baysian conditional probability, and chaos and order. To the authors knowledge, the relationship of p_s in the coefficients in the bakers transformation to temperature had never been shown before for technology transfer, or software evolution.

The social structure, as defined by Rogers, is not directly addressed in the model, but rather would be addressed by a social network analysis method such as Burt's structural holes. Another approach is to look at the money distribution and exchange between research organizations. Their revenue income, money, would be exchanged with the environment. We might say, making a simplifying assumption that the only

major stimuli is funding, that the funding distribution by performer bands per capita might give insight into a stimuli aspect (heat). This follows from studies of the economy using statistical mechanics (Dragulescu 2000).

The model we have described here is analogous to those used for working with mass flows, entropy, pressures and temperatures. There is no discussion of the strength of the materials (e.g. social structure[25]) or the details of the implementation of the end product – the engine. This is as it should be.

We have laid out the fundamentals. The size of the nodes, the production ($\dot{m}$, message flow), and even hidden in here are the elements of pressure (messages per node), and temperature, $1/T$, the reciprocal of the uncertainty slope $\partial S/\partial n$, the *coldness.* (Fraundorf 2000), (Schroeder 2000). Massieu (1869) provided the start point for the generalized ensemble relations with the Massieu-Plank functions for statistical entropy, (Műnster 1969), (Planes 2002). We see the a set of entropic potential formulations for technology transition dynamics are now available to the community.

We now have available a method to measure a technology's temperature. Temperature represents the propensity for a system to share properties, information in this case. We have worked with some basic tools and used the quantitative version of the zeroth law. This could apply to many aspects of software systems, even indices in data structures when properly constructed. The theory under all of this need not apply only to energy, or information. It also applies to unequilibrated systems sharing conserved quantities (money for example), if the only prior information we have is how the multiplicity of ways that quantity can be distributed depends on the conserved quantity to begin with! (Planes 2002)

When we have other kinds of information, such as knowledge of a systems temperature (the slope of the uncertainty) but not its total information, then the broader class of maximum entropy strategies in statistical inference (e.g. the canonical and grand ensembles) predict the distribution of outcomes we can expect as well.

---

[25] The veracity (social capital, per Burt) of the publisher (Carnegie Mellon vs Podunk Community College) is left to the designer of the engine desired, and the implementers who fabricate the engine.

1.   Technology Transition Engine

Now that we have established the basic relationships of the *TechTx Entropy* models, lets put it in the framework of a system. We can put it all together as an evolutionary, technology transfer system that has probabilistic effects at the macro level and deterministic, dynamical effects at the microstate level. We have to the tools to analyze a program and represent it as an engine.

2.   Control Volume

It is useful to define a control volume that is typical of the system Figure V-1. In a traditional continuous system in a physical world, a control volume identifies boundaries of the system. In such a continuous system, say an engine, a mass flows a distance and contributes to the work performed. It is not unusual to partition up a continuous control volume into stages, e.g. a compressor, a combustor, a diffuser. As the mass *m* flows from stage to stage, we can consider it a state transition of the system and locally of the nodes (compressor, combustor, diffuser). There are *n* masses flowing, each one unique, so the system and the nodes take on different states for the complete elaboration of the mass-node states combinations. For now, let's look at all three nodes in the message model with the mass replaced by the message moving through the control volume. This causes both local and system level state transitions.

Control Volume
Continuous and Discrete Example
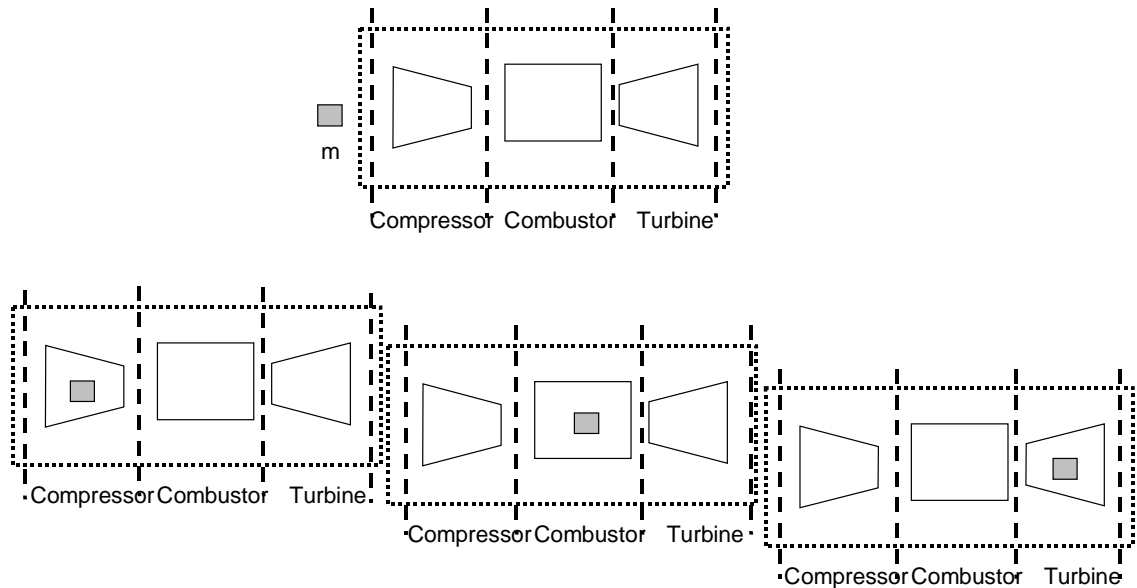
The nodes transition to a different state as the mass *m* is present.
This is the analog of a discrete state machine in a continuous system

Figure V-1   Illustration of a Control Volume -- a Continuous System or as a Discrete State Machine

Similarly, in this discrete state machine, we have drawn the boundaries around the three nodes. Full elaboration of all of the messages *(m)* states within the control volume would represent all the possible states of the bounded system.   With this, we can represent an individual interaction, an organizational interaction or even a macro technology transfer system such as the economy.

3.   State and Cycle Diagrams

These technology transition dynamics tools permit us to engineer a solution to get maximum efficiency out of our resources.   Let's examine some of the state diagrams and system quantities in Figure V-2.

System Quantities
Q, H, W
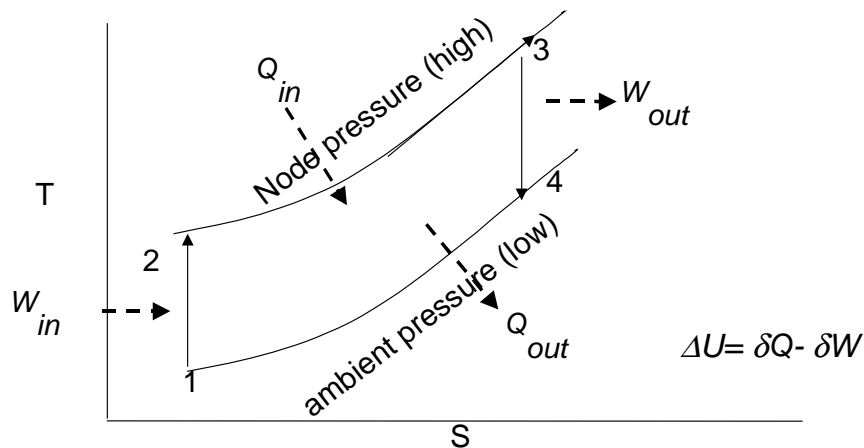
Figure V-2 Technology Transfer State Diagram, System Quantities

This section will develop the relationships of a temperature entropy (T-S) diagram familiar to mechanical engineers when performing engine thermodynamic cycle analysis. We suggest that what are the conditions for moving up from one "pressure – temperature – entropy" state (numbered 1-4) to another.

Here we have a process depicted in macro state space that originates at point 1 with T_{lo}, P_{lo}, S_{lo} which are the ambient temperature and pressure of the surroundings, a reservoir. In a sense, we see this as work, energy or heat. In technology transfer dynamics, we can think of this as effort, which is added to the system, yielding "energetic" messages. We see an *isentropic*, (constant entropy) compression as the system moves along the path 12 to T_2, P_{hi}, S_{lo}. This says the temperature is increasing because some effort is being done to reduce the volume in which the interaction between

entities occurs. More occurrences of existing terms consistently show up in messages. Terms are combined to get to concepts that are more powerful. While there may be less volume, fewer nodes, the message term content has higher density. In the model proposed, there would be fewer nodes, but doing very intense research, i.e. producing much high quality messages. They closely interact and publish messages generally within the confines of the system.

During the progression form state point *2* to *3,* energy in the form of effort is added at a constant pressure. Entropy, S_{lo} increases to S_{hi}. Think of this as a demonstration. No new basic research is being performed, the science is being scaled up and loaded with a lot of energy that will make it attractive to consumers. This occurs when the technology is *diffused* from state *3* to *4.* A high pressure, concentrated set of messages escapes into a In order for this to happen the message entities must some how move to a bigger volume, must some how escape. This is where work is taken out, as products are delivered to a market (ambient). This is shown as a constant entropy line, which a rapid drop from T_{hi},P_{hi},S_{hi}, at state point *3,* to state point *4, T_4,P_{lo},S_{hi}.* Work, in thermodynamic terms, is represented by extensive property rate changes. For example,

$$W = \dot{n}C_p(T_3 - T_4) \qquad\qquad (5.7)$$

Where *W* is work (product) yield, $\dot{n}$ is message flow, C_p is the specific heat at constant pressure, and *T* is the temperature. While the technology transfer dynamics doesn't have foot-pounds per se, it does have a state change per time step, and terms and sets of sets of terms are the extensive property. The sets can even have "weights" based on the primitive terms in the set, or the q-level.

Figure V-3 illustrates the *T-S* cycle diagram for Ada. We mist recognize that this was not an "engineered" system, yet we can still see the faint trace of the cycle of Figure V-2. Note the super imposed cycle diagram. We recognize that we can not achieve constant entropy, so the first state transition move up and to the right. This is when the early researchers, innovators and early adopters, are at work. The next state change is the chord which moves off to the right, might suggest that early adopters and early majority are adopting and performing experiments, withstanding some pressure above ambient,

and demonstrating internally and externally. In fact it looks like the is a steady increase in pressure, until the maximum when the system starts to diffuse, the state transition that drops off and toward increasing entropy, but lower temperature. Lower temperature implies lower pressure. The rectangle represents the ideal cycle, the Carnot cycle. The maximum efficiency is limited by the ration of T_{lo}/T_{hi}.

Temperature - Entropy (T-S)
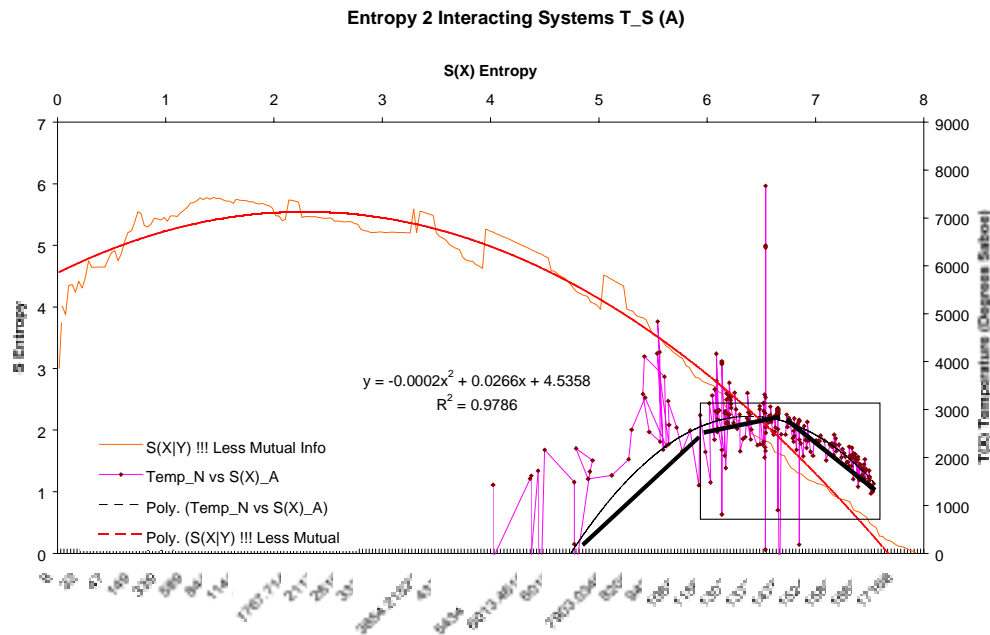
Figure V-3 Temperature Entropy Diagram – Ada

The research tied together fundamental elements underlying technology transition. Currently, systematic techniques for assessing macro mechanisms for transferring software-engineering technologies has been thoroughly reviewed and systematized. This dissertation developed the fundamental elements of an *industrial model* of a *software technology transition engine*. The mechanisms developed utilizing information theory, communication theory, chaos control theory, and learning curve principles. The combination of those scientifically sound mechanisms provides a basis for assessing, and / or prescribing a portfolio of technologies and the implementing macro infrastructure.   Linkages to lower level models and implementation methods are

provided.. This research provides the *engineering framework for a practical method for a program manager to establish a high capacity transition channel, which accelerates technology maturation and insertion*. Data samples assess the following technologies: software technology transfer, Ada, Java, abstract data types, rate monotonic analysis, cost models, software standards, software work breakdown structures. Also included is an extensive annotated bibliography on software technology transfer and related references, and a bibliography including related material from philosophy, psychology, math, physics, thermodynamics, management, economics, game theory, technology transfer, software engineering, and systems engineering.

The application of foundational relationships permits a development of a software technology transition engine.

Finally, it is left to the community to determine whether this is satisfactory to support the following logic:

- since we should be able to accept that a process is just a program (Osterweil 1987) and

- software can represent the program, and

- the engine is the representation of a process that was based on axiomatic and logical transfers from established science and engineering (physics and thermodynamics)

- The basic elements of the ***physics of software*** have been developed

A broad area of future research is outlined in the next section.

THIS PAGE INTENTIONALLY BLANK

VI.   IMPLICATIONS FOR FUTURE RESEARCH.

The research explored the use of entropy in information theory. Great effort was put into ensuring that units (as in dimensional units) are consistent across the various analysis techniques using measures. The unit analysis drives toward statistical inference techniques. For example, the common unit for length is measured in informational units and related to various distributions.   This section suggests areas of future research in the areas of:

- Development of "engine" design and analysis, applicable to technology transition, evaluation and risk and general enough to be applied to the evolutionary software development process, and software itself.

  - Application of the entropy metric to the evolutionary software development process.

  - Linkage of messages in the software development process to the software application.

  - Analysis and linkage of software to the information theoretic, and dynamical systems, the dynamical system linkage is only now available as the result of this research.

- Development of a complexity metric for software, which computes the temperature from both the structure (information theoretic) and the flow (dynamical micro model)

- Learning curve relationship of performance and entropy

- Exploration of the use in molecular and biologically inspired computing, so that we no longer "program" software rather we grow it.

- Developing the relationship to quantum mechanics and exploring the possibility of an "information field theory". This would explore maximum

entropy as the underpinning construct that governs physical gravity, or the tendency for bodies to attract i.e. desire mutual information through the need for correlation of various properties.

- Finally, explore the implications of relationships of software physics to a quantum theology, and the true mysteries of the universe.

A. TECHNOLOGY TRANSITION ENGINE

The work drives toward an "engine" that has a simple control mechanism, just as one might imagine, -- a gas pedal or throttle. This means all of the various components are in balance (there is a predicate relationship at the boundaries that must be satisfied) and represent a dynamic system. The engine also is affected by the economy (the environment) at the control volume boundary of the system. Let's suggest a metaphor for additional research. Assume that the technology transfer engine is like a jet engine, the amount of thrust it can produce from the ejected (and conserved) quantities is very much a function of the thermodynamic design of the engine. This is the bulk of the effort in the model; however, if the jet's diffuser ejects at a speed relative to the engine's forward motion and high altitude jet stream, the total speed is some aggregation of all of these effects. Since we wish to predict, with some confidence, whether a technology will arrive at a given time, these "macro economic - environmental" factors must be represented in the model.

There is a juicy direction for further research further developing that metaphor of thermodynamics and information theory. From that point of reference, one can envision a second law analysis, i.e. focusing on the inefficiencies. Those inefficiencies establish the requirements to the technology base in a "problem oriented", "requirements pull" approach. Viewing this in the thermodynamic cycle metaphor, imagine the waste heat going out the exhaust (i.e. scrap and rework in the software development process) being redirected to preheat or regenerate the input into the cycle (i.e. guide the research agenda and focus on the heavy payoff opportunities).

Temperature Entropy Diagram
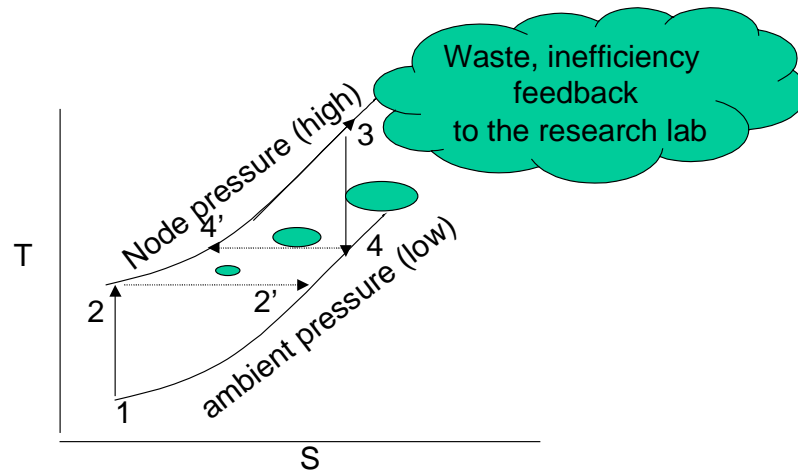
Figure VI-1.Technology Transition Engine Temperature Entropy Diagram.

Future research should experiment and calibrate the specific heats of various technologies and software. Ambient temperature should be calibrated for general regions of research in technology domains.

A software technology transition (Tech Tx) engine could be analyzed with the tools (Temperature, Pressure, entropy, messages, and specific heat) developed.

It should be argued that such an engine, which pumps technologies to the user community, should have certain properties. The object would be to design an efficient, i.e. the maximum amount of work product should get to the goal of insertion with the minimum amount of resources consumed and wasted. It is suggested that the use of a cycle diagram, familiar to physicists, mechanical engineers and thermodynamicists, could be used to evaluate the efficiency of the technology transfer engine. This approach is similar to a Carnot cycle analysis using state points of entropy, temperature, and pressure. Related to analysis of the engine suggests areas for additional work: the notion of

"squaring the Carnot cycle"; the Second Law Analysis, a description of the TechTx engine in terms of evolutionary software development process; and identification of software development *entropy* metric. Further, since this research has based its foundation on physics and thermodynamics, we now have the full richness of those disciplines potentially available. This will permit building on existing theory in these areas with the language familiar to the scientist and engineer.

With such tools, a decision-maker would be able to determine the confidence that a technology or group of technologies will arrive on at a given time frame within a certain confidence limit. For example, a program might expect a portfolio of technologies to arrive by year 06 with an 80% certainty, but the model might show that in 06, there is only 60% certainty of being available using the current trends. (See Figure VI-2). The desired 80% certainty would not be available until 08. If the technology is not predicted to arrive as required, the model will point to the areas for remedy with a prescriptive solution as to how to *organize, train and equip* in order to change the confidence of arrival.

Program Office Use for Risk Assessment and Rx

Example:
Program Office Wants
by 06 with 80% certainty

Analysis indicates 08

What nodes / programmatics
need to be put into place to
shift curve to left?

From desired system curve
Algebraically solve for node response curves(s)

Determine how many and parallel / serial

80%

60%

06   08
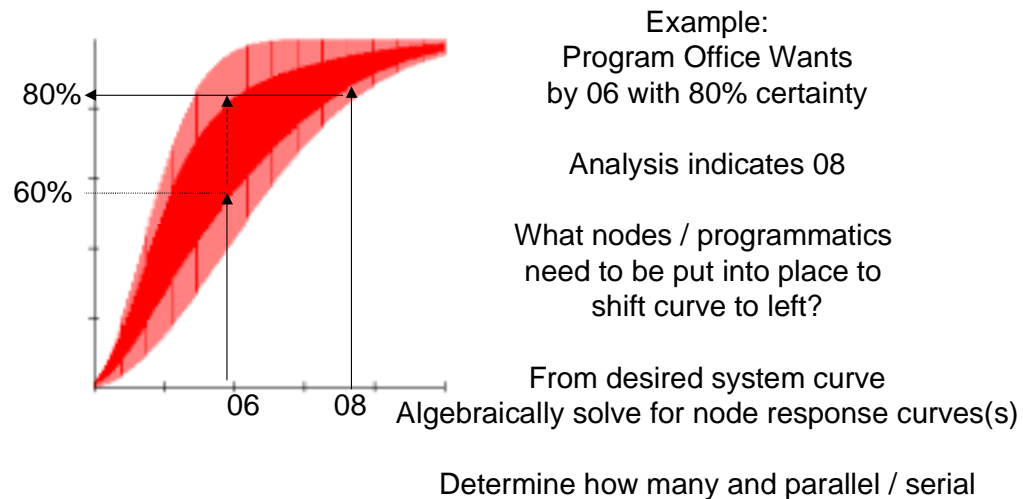
Figure VI-2.   Model Usage in Program Office Technology Risk Assessment.

While this research developed the general relationships of properties, the application code to do the analysis was limited to the needs of generation data to validate the relations.   The application macros were written to easily be incorporated in to Microsoft Office applications.   A user interface that permits a program manager, or technology policy maker to perform "what if scenarios" would be most useful.

The concept of entropy for a software technology transfer process is defined. This entropy concept is also adapted to meet the character of an evolutionary software development process.   From this pivot point with the intensive properties such as temperature and heat capacity -- now expressible in information units, a model can be developed for the software technology transition engine. The model developed, herein, has the features of a communication and control system theory. It accommodates mixing effects, chance, and the maturity of the individual organizational units to reflect a

learning organization unit consisting of people and machines. This was done with the separation of microscopic issues from the macroscopic using the analysis of stable dynamical systems, and relating the properties of these system properties to the dynamics of the system nodes.

B. SOFTWARE COMPLEXITY METRIC BASE ON ^{o} DEGREES SABOE

As we saw in Chapter III, we have a relationship to the Weibull probability distribution function.

$$P(q_i) = e^{-\left(\frac{q_i+\gamma}{\beta}\right)^{\alpha}} \quad \text{a Weibull Distribution} \quad (6.1)$$

The Weibull is used in other research (Nogeria 2000) to address a number factors that effect an evolutionary process. In Nogeria's case, it was used to model the requirements volatility, efficiency of the performers in the process and the size of a software artifact indicated by a complexity. It should also be noted that in that study, the independent variable was time. In this case, we are addressing messages, q_i, or the structure of the artifact to determine a measure of complexity (temperature). The number of messages processed in a time step can be converted to time as an independent variable with some mathematical manipulations. This can be related to the learning curves.

There was some difficulty in addressing complexity, in that research. The use of microstates of an alphabet, and temperature may contribute to advancing related research efforts. In this case, we might let the *x*-axis shift of the Weibull, $\gamma=0$, and see that $\alpha \approx 1$.

There is a close connection to Halstead metrics as stated earlier. Halstead metrics can be easily connected to the temperature. He determined the alphabet of operators and operands. Looking carefully at his equations he is very close to using entropy as a metric, but just misses the connect by a simple division.

He defined the program volume *V* by

$$V = N \log_2 \eta \quad \text{(Halstead 1977, p19, eqn 3.1)}$$

Where *N* is related to total usage of operators and operands. Defining each operator and operand as a term, these are the instances counts (*n)* in this dissertation. The number of distinct operators and operands (terms in our terminology) is his η. Had he not used the actual numbers, but rather summed the probabilities of occurrence and log of

the probabilities he would have had Shannon's equation. $S = -\sum_{x \in \Xi} p(x)\log_2 p(x)$, and he would have had the (Saboe) entropy metric for the software.

He related input data streams and program levels. Linking temperature, and entropy as defined in this dissertation to software volume, and length metrics of Halstead will bring the ability to quantify abstraction using the entropy contribution approach.

Where each meta level, partition, band or module i, provides a contribution, C_i, to the total population entropy. The local entropy S_{H_i}, can be scaled based on the multiplicity Ω_i of terms in the band to the multiplicity Ω of terms in the population. Similar to the equations we introduced earlier. With the total population's entropy is the sum of the contributions.

$$S_{H_{world}} = \sum_{i=1}^{n\_bands} C_i \qquad (6.2)$$

$$\text{where } C_i = \frac{|\Omega_i|}{|\Omega|} S_{H_i} + \frac{|\Omega_i|}{|\Omega|}\log\frac{|\Omega|}{|\Omega_i|} \qquad (6.3)$$

Halstead was instead limited to programming language view. Her we can start to deal with abstraction and complexity, a subject the is careful to say is not addressed.

Halstead did not use the notion of q-levels. This can make a great difference in the power of his metrics and provides one of the missing ingredients, temperature.

The linkage of the dynamical equations can be shown through McCabe's metric, cyclometric complexity.

Going through Halstead metrics, we can get to the Stroud number. This is related to tasks (decisions) per time step. That linkage will be suggested as an area of future research in the learning curve section.

C.   *TECHTX ENTROPY LEARNING CURVE* MODEL, MICRO LEVEL DATA ANALYSIS

1.   Nodal Performance Data

As we saw in Chapter III, in order to get to the right level of granularity, the performing organization nodes criteria and bands are assessed and presented. The distribution of the performance index for the complete data set is shown in four bands. The capacity performance index over time is shown for each of the bands. This represents the best that the band can do (on average) at the time of performance. The entropy is allocated to the performing nodes (affiliated organizations) using a per capita rate in a band.

The output entropy is allocated from the message to individual performers from the empirical data. This micro level is then summed up and allocated to the to the affiliated organizational level. The organizations are banded based on a distribution of the cumulative number of published messages. This accumulation of experience is from the beginning of the data set to the time step at the performance time step. In this case, that is 22 years. An example of the distribution is shown in Figure VI-3. We see the standard cast of high performing nodes. These are world class research organizations (with the Naval Postgraduate School in the top 15 of over 1500 organizations).

Productivity Distribution (Ada)
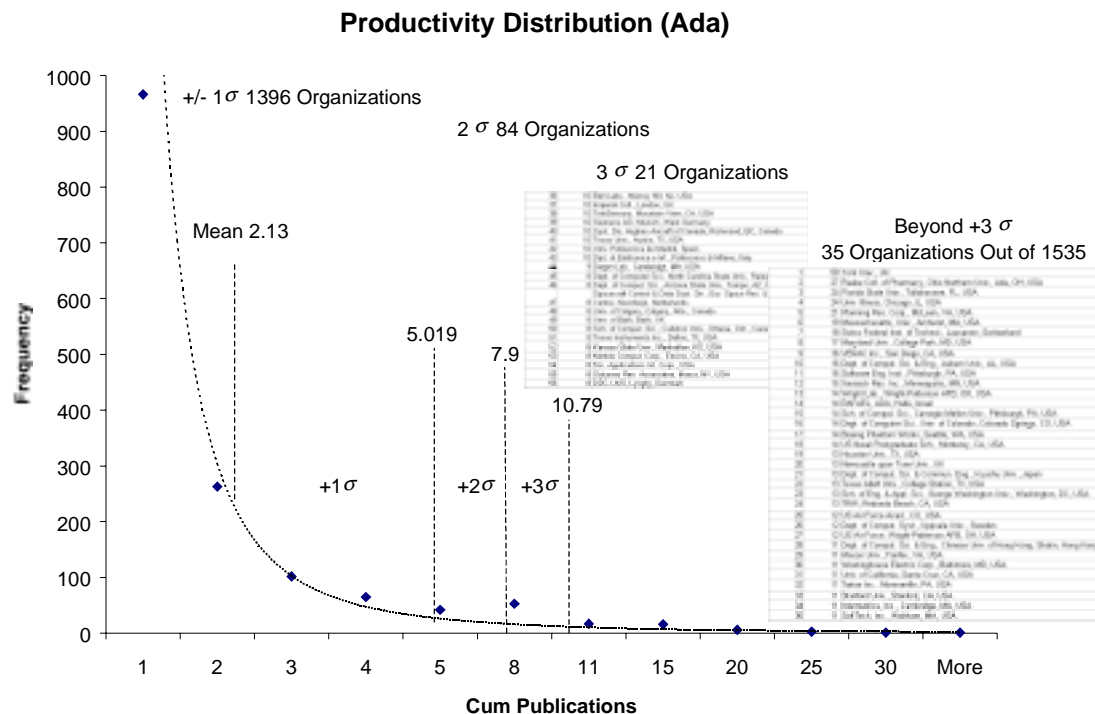
Figure VI-3 Performing Organization Distribution Bands at End of Data Set

D. MOLECULAR AND BIOLOGICALLY INSPIRED COMPUTING

Molecular and biologically inspired computing could possibly build from the relationships developed in this research. In the future, it is possible that we will be unable to "program" molecular computers as we do today. We will want to grow software. The software will likely compute similar to biological systems that evolve. They likely will use patterns and associations, and move in the direction of least resistance, and maximum potential. The model development in Chapter III addressed relationships, changes, and lock in effects for the technology in question. It may be able to be adopted for the more general class of evolutionary system.

One could synthesize attractors and repellors (sources and sinks) to guide the process. This is similar to the strategy, which might develop macro economic - environmental effects that drive technologies from an evolutionary growth aspect.

The research proposed a software technology transition cycle analysis approach. This permits analysis of various approaches for policy and investment trades. Tools that build on this analysis approach can help identify leverage points and opportunities to accelerate progress with a repeatable and rigorous approach.

In this type of environment, we can make the relationship of the macro/microscopic connection explicit. The work should provide an axiomatic development for a second law analysis -- think of this as analyzing the inefficiencies, which in turn provides a mechanism for feedback.

Future work addresses implementing the model in an organization, writing policy to enable the realization of the model and experimentation to validate the theory.

LIST OF REFERENCES

(AAAS 1972) "Liebig and After Liebig: A Century of Progress in Agricultural Chemistry," *American Association for the Advancement of Science*, Bulletin 16, 1972

(Abbott 1989) Abbott, Michael M. and Nan Ness, Hendrick*, Theory and Problems of Thermodynamics*, McGraw-Hill, New York, 1989.

(Allen 1977) Allen, Thomas J., *Managing the Flow of Technology*, MIT Press, Cambridge, MA, 1977.

(Allen 1983) Allen, Thomas J., Diane B. Hyman, and David L. Pinckney, "Transferring Technology to the Small Manufacturing Firm: A Study of Technology Transfer in Three Countries," *Research Policy*, 12, pp. 199-211, 1983.

(Anderson 1981) Anderson, John R., ed., *Cognitive Skills and Their Acquisition*, Lawrence Erlbaum Associates, Hillsdale, N.J., 1981.

(Asher 1956) Asher, H., "Cost-quantity Relationships in the Airframe Industry," Rand Corporation, Santa Monica, CA, Rep. R291, 1956.

(Baker 1990) Baker, Gregory L. and Gollub, Jerry B., *Chaotic Dynamics an Introduction,* Cambridge University Press, 1990.

(Barndorff-Nielsen 1993) Barndorff-Nielsen, O.E., Jensen, J.L, and Kendall, W.S., *Networks and Chaos – Statistical and Probabilistic Aspects*, Chapman & Hall, London, 1993.

(Basili 1994) Basili, Victor R., Selby, Richard W., and Hutchens, David H., "Experimentation in Software Engineering" The Institute of Electrical and Electronics Engineers, Inc. 1994

(Basili 1994a) Basili, Victor, Geen, S., "Process Evolution at the SEL", *IEEE Software*, 11: (4) pp. 58-66, July 1994.

(Bayes 1763) Bayes, Thomas, Rev., "Studies in the History of Probability and Statistics: IX, Thomas Bayes's Essay Towards Solving a Problem in the Doctrine of Chances", Barnard, G. A, *Biometrika*, Vol 45, Issue 3 / 4 Dec. 1958, pp.293-315.

(Behnke 2001) Behnke, Matthew, *Development Of An Automation Tool To Compute The Cumulative Entropy Of Datasets*, Bachelor of Science in Computer Science Thesis Kettering University, United States Army Tank Automotive & Armaments Comma, October 2001.

(Berniker 1991) Berniker, E., "Models Of Technology Transfer: A Dialectical Case Study," *Proceedings of the IEEE Conference*: The New International Language, July 1991.

(Berzins) Berzins, V. and Luqi, *Software Engineering with Abstractions*, Addison Wesley Publishing, 1991.

(Blum 1996) Blum, B.I., *Beyond Programming to a New Era of Design*, Oxford University Press, 1996.

(Boehm 1988) Boehm, Barry W., "A Spiral Model of Software Development and Enhancement," *IEEE Computer*, (21,5), pp.61-72, 1988.

(Boehm 1989) Boehm, Barry, and Ross, R., "Theory W Software Project Management: Principles and Examples," *IEEE Transactions of Software Engineering*, July 1989, pp. 902-916.

(Boehm 1998) Boehm, Barry, Egyed, A., Kwan, J., Port, D., and Shah, A., "Using the WinWin Spiral Model: A Case Study," *IEEE Computer*, July 1998, pp. 33-44.

(Boehm 1999) Boehm, Barry B. TBD CeBase…

(Boehm 2000) Boehm, Barry and Victor Basili, "Gaining Intellectual Control Of Software Development", *IEEE Computer*, May 2000, pp. 27-33.

(Boehm 2001), Boehm, Barry (USC, Center for Software Engineering), Bill Scherliss (Carnegie Mellon University), Peter Kind (LTG retired) (Institute for Defense Analysis), Tony Jordano, (SAIC), Michael Saboe, (US Army Next Gen Software),

Department of Defense Software Engineering Science and Technology Summit, University of Southern California, Los Angeles, California, August 7, 2001.

(Brown 1992) Brown, R., "Generalizations of the Chua Equations," *International Journal of Bifurcation and Chaos* 2(4), 1992.

(Brown 1992a) Brown, R. and Chua, L., "Chaos or Turbulence?" *International Journal of Bifurcation and Chaos* 2(4), 1992.

(Brown 1993) Brown, R. and Chua, L., "Dynamical Integration," *International Journal of Bifurcation and Chaos* 3(1), 1993.

(Brown 1993a) Brown, R, Chua, L.,. & Hamilton, N., "Fractals in the Twist-and-Flip Circuit," *Proceedings of the IEEE" Special Issue on Fractals in Circuits*, October 1993.

(Brown 1993b)   Brown, R. "From the Chua Circuit to the Generalized Chua Map," *IEEE Transactions on Systems and Circuits,* Special Issue on Chua's Circuit, 1993.

(Brown 1993c)   Brown, R. and Chua, L., "Dynamical Synthesis of Poincare Maps," *International Journal of Bifurcation and Chaos* 3(5), 1993.

(Brown 1995) Brown, R., "Horseshoes in the Measure Preserving Henon Map," *Ergodic Theory and Dynamical Systems*, 1995.

(Brown 1996)   Brown, R. and Chua, L, "Clarifying Chaos: Examples and Counterexamples," *International Journal of Bifurcation and Chaos* 6(2), 1996.

(Brown 1996a) Brown, R. and Chua, L., "From Almost Periodic To Chaotic: The Fundamental Map," *International Journal of Bifurcation and Chaos* 6(6), 1996.

(Brown 1997)   Brown, R. and Chua, L., "Chaos: Generating Complexity From Simplicity," *International Journal of Bifurcation and Chaos* 7(7), 1997.

(Brown 1998) Brown, R. and Chua, L., "Clarifying Chaos II: Bernoulli Chaos," *International Journal of Bifurcation and Chaos* 8(2), 1998.

(Brown 1999)   Brown, R. and Chua, L., "Clarifying Chaos III: Stochastic Processes" "International Journal of Bifurcation and Chaos 9(5), 1999.

(Brown 1999a) Brown, R., "On Solving Nonlinear Functional, Finite Difference, Composition, and Iterated Equations," to appear in *Fractals*, 1999.

(Brown 2000) Brown, Ray, "Private Communication," *Toward a Control Theory for C^3 Systems*, Research Fellow, Raytheon Systems Company, Falls Church, VA 22042, September 2000, through May 2001.

(Burt 1992) Burt, Ronald S., "The Structure of Competition", Chapter 2 in Nohria and Eccles (ed.), *Networks and Organizations: Structure, Form, and Action*, Harvard Business School Press, Boston, Massachusetts, 1992, and in the book *Structural Holes*, Harvard University Press, 1992.

(Buxton 1991) Buxton, J.N. and Malcolm, R., *Software Technology Transfer*, pp.17-23, 1991.

(Carr 1946) Carr, G.W., "Peacetime Cost Estimating Requires New Learning Curves," *Aviation*, vol. 45, pp. 75-77, 1946.

(Çengel 1989) Çengel, Yunus, and Boles, Michael, *Thermodynamics: An Engineering Approach*, ISBN 0-07-010356-9, McGraw-Hill, 1989.

(Chase 1981) Chase, William G. and Ericsson, Anders K., "Skilled Memory," in Anderson, John R, ed., *Cognitive Skills and Their Acquisition*, Lawrence Erlbaum Associates, Hillsdale, N.J., 1981.

(Cover 1991) Cover, Thomas M. and Thomas, Joy A., *Elements of Information Theory*, John Wiley and Sons, Inc., N.Y., 1991.

(DeJong 1957) DeJong, J.R., "The Effects of Increasing Skill on Cycle Time and Its Consequences for Time Standards," *Ergonomics*, pp. 51-60, 1957.

(Dragulescu 2000) Dragulescu, Al, and Yakovenko, V.M., "Statistical Mechanics of Money," The European Physical Journal B, 17, pp. 723-729, 2000.

(Dretske 1988) Dretske, Fred I., *Explaining Behavior, Reasons in a World of Causes*, MIT Press, 1988.

(Dretske 1981)   Dretske, Fred I., *Knowledge and the Flow of Information,* Bradford Books, 1981.

(DSB 2000) *Defense Science Board, Defense Software*, Nov. 2000.

(Elskens 1986) Elskens, Y. and Prigogine, I., "From Instability to Irreversibility," Proceedings, National Academy of Sciences, USA, Physics, Vol. 83, pp. 5756-5760, August, 1986.

(Eveland 1990) Eveland, J.E. and Tornatzky, L.G., "The Deployment of Technology," in *The Processes of Technological Innovation*, L.G. Tornatzky and M. Fleischer, editors, pp. 117-148, Lexington Books, Lexington, Massachusetts, 1990.

(Farmer 1983)   Farmer, J. Doyne, "The Dimension of Chaotic Attractors," *Physica* 7D, pp. 153-180, North-Holland Publishing Co., 1983.

(Fast 1962) Fast, J.D., *Entropy*, McGraw-Hill, New York, 1962.

(Fichman 1993)   Fichman, Robert G. and Kemerer, Christ F. "Adoption of Software Engineering Process Innovations: The Case of Object Orientation," *Sloan Management Review*, Winter, pp. 7-22, 1993.

(Fichman 1994) Fichman-RG; Kemerera-CF, "Toward A Theory of the Adoption and Diffusion of Software Process Innovations," in *Diffusion, Transfer and Implementation of Information Technology*, in Levine, Linda, ed., proceedings of the IFIP TC8 Working Conference on Diffusion, Transfer and Implementation of Information Technology, Software Engineering Institute, Carnegie Mellon Institute, Pittsburgh, PA, North Holland, 1994.

(Forrester 2000) DoD Evolutionary Acquisition Workshop, September, 2000.

(Fowler 1990) Fowler, P., "Technology Transfer As Collaboration: The Receptor Group," *Proceedings of the 12th International Conference on Software Engineering* pp. 332-333, IEEE Computer Society Press, Nice, France, U.S., 1990.

(Fowler 1992) Fowler, P. & Levine, L.,"Toward A Problem Solving Approach To Software Technology Transition," in J. Van Lemwen (Ed.), *Proceedings of the IFIP 12th*

World Computer Congress, vol., pp. 57-64, Madrid, Spain, The Netherlands: North Holland, Elsevier Science Publishers, 1992.

(Fowler 1992A) Fowler, P. & Maher, J., "Foundations For Systematic Software Technology Transition," *Software Engineering Institute Technical Review* '92, pp. 1-32, 1992.

(Fowler 1994) Fowler, Pricilla and Levine, L., "From Theory to Practice: Technology Transition at the SEI," *IEEE Proceedings of the Twenty-Seventh Annual Hawaii International Conference on System Sciences*, pp. 483-497, 1994.

(Fraundorf 2000) Fraundorf, P., "Heat Capacity in Bits," xxx.LANL.gov/cond-mat/9711074 v2, October 1, 1999, rev. October 30, 2000.

(Garrod 1995) Garrod, Claude, *Statistical Mechanics and Thermodynamics*, Oxford University Press, 1995.

(Gibbs 1928) Gibbs, J. Willard, *The Collected Works of J. Willard Gibbs*, Ph.D., LL.D, Volume 1, *Thermodynamics*, Longmas, Green and Co, New York, 1928.

(Gibson 1989) Gibson, J.E.; Heilig, V.K, *The Challenge Of Technology Transfer (Software Engineering Curriculum),* Fairley, R. and Freeman, P. editors, Carnegie-Mellon Univ., 1989.

(Grable 1994) Grable, Ross, "A State Based Software Entropy Metric", US Army Missile Command, Personal communication, November 11, 1994.

(Graettinger 2000) Discussion with Dr. Caroline Graettinger, Software Engineering Institute at the *Evolutionary Acquisition Workshop* for the DoD, Washington DC, September 2000.

(Graettinger 2001) *DoD Software Collaborators Workshop*, University of Southern California, Los Angeles, CA, February, 2001.

(Graettinger 2001a) *Department of Defense Software Engineering Science and Technology Summit*, University of Southern California, Los Angeles California, August 7, 2001.

(Gulliksen 1934)   Gulliksen, H., "A Rational Equation of the Learning Curve Based on Thorndike's Law of Effect," *Journal of General Psychology*, 11, pp. 395-434, 1934.

(Halstead 1977)   Halstead, Maurice H., *Elements of Software Science*, Elsevier North Holland, Inc., New York, 1977.

(Hanakawa 1998) Hanakawa, Noriko; Morisaki, Syuji; Matsumoto, Ken-ichi, "A Learning Curve Based Simulation Model for Software Development," pp. 350-359, Nara Institute of Science and Technology, IEEE, 1998.

(Hargadon 1997)   Hargadon, Andrew and Sutton, Robert I., *Technology Brokering and Innovation in a Product Development Firm*, Cornell University, 1997.

(Haskins 1927)   Haskins, C., *The Renaissance of the Twelfth Century*, Harvard, 1927.

(Hirshliefer 1994) Hirshliefer, Jack and Riley, John G., *The Analytics of Uncertainty and Information*, Cambridge University Press, 1992.

(Huang 1963)   Huang, Kerson, *Statistical Mechanics*, John Wiley & Sons, New York, 1963.

(Jaakkola 1995)   Jaakkola, Hannu, "Comparison and Analysis of Diffusion Models," p 65-82, 1995.

(James 1890)   James, William, "The Principles of Psychology," *Great Books of the Western World*, Book 53, M. Adler, Associate Editor, *Encyclopaedia Britannica*, Inc., ISBN 0-85229, pp. 163-9, Chicago, 1952.

(Jaynes 1957)   Jaynes, E.T., "Information Theory and Statistical Mechanics," *Physical Review*, Vol. 106, Number 4: pp.620-630, May 15, 1957.

(Jaynes 1957a) Jaynes, E.T., "Information Theory and Statistical Mechanics II," *Physical Review*, Vol. 108, Number 2, pp. 171-190, October 15, 1957.

(Jovanovic 1999)   Jovanovic, Vladan, and Schoemaker, Dan, *Engineering A Better Software Organization*, Quest Publishing, Detroit-Ann Arbor, 1999.

(Katz 1961)   Katz, Elihu, "The Social Itinerary of Technology Change: Two Studies on the Diffusion of Information," in Wilbur Schramm (ed*.) Studies of Innovation and of Communication to the Public*, Stanford, California, Stanford University, Institute for Communication Research, and *Human Organization*, 20 pp.70-82, 1962.

(Knecht 1974)   Knecht, G.R., "Costing, Technological Growth and Generalized Learning Curves," *Operations Research Quarterly*, vol. 25, no.3, pp. 487-491, 1974.

(Kolmogorov 1956)   Kolmogorov, Andrei N., "On the Shannon Theory of Information Transmission in the Case of Continuous Signals," *IRE Transaction on Information Theory*, December 1956, presented at 1956 Symposium on Information Theory at Mass. Inst. Tech, Cambridge, MA, September 10-12, 1956.

(Korn 1961)   Korn, G.A. and T.M. Korn, *Mathematical Handbook for Scientists and Engineers*, pp.18.4.12, McGraw-Hill, New York, 1961.

(Kreyszig 1993)   Kreyszig, Erwin, "Advanced Engineering Mathematics" book, Seventh Edition, John Wiley & Sons, Inc., 1993.

(Kulch 1972)   Kulch, W. "Entropy of Transformed Finite State Automata and Associated Languages", in *Graph Theory and Computation*, R.C. Read, ed. Academic Press, New York, 1972.

(Kwon 1987)   Kwon, T. H. and Zmud, R.W., "Unifying the Fragmented Models of Information Systems Implementation" in *Critical Issues in Information Systems Research,* eds. J.R. Boland and R. Nirshheim, John Wiley & Sons, New York , 1987.

(Langley 1981)   Langley, Pat and Simon, Herbert A., "The Central Role of Learning in Cognition," in Anderson, John R, ed., *Cognitive Skills and Their Acquisition*, Lawrence Erlbaum Associates, Hillsdale, N.J., 1981.

(Leagans 1979) Leagans, J. Paul, *Adoption for Modern Agricultural Technology by Small Farm Operators: An Interdisciplinary Model for Researchers and Strategy Builders,* CIAM 69, NTIS PB82-234527, Cornell University Department of Education, Ithaca, New York, 1979.

(Leonard-Barton 1988) Leonard-Barton, D., "Implementation Characteristics of Organizational Innovations," *Communication Research* 15, pp. 603-631, 1988.

(Levine 1994) Levine, Linda, ed., *Proceedings of the IFIP TC8 Working Conference on Diffusion, Transfer and Implementation of Information Technology*, Software Engineering Institute, Carnegie Mellon Institute, Pittsburgh, PA, North Holland, Amsterdam, London, New York, Tokyo, 1994.

(Levy 1965) Levy, "Adaptation in the Production Process," *Manage.Sci*, vol. 11, no. 6, pp. 136-154, 1965.

(Lewis 1981) Lewis, Clayton , "Skill in Algebra," in Anderson, John R, ed., *Cognitive Skills and Their Acquisition*, Lawrence Erlbaum Associates, Hillsdale, N.J., 1981.

(Li 1993) Li, Ming and Vitanyi, Paul, *An Introduction to Kolmogorov Complexity and its Applications*, ISBN 0387-94053-7, Springer-Verlog, New York, 1993.

(Lienhard 2000) Leinhard, John H., *The Engines of Our Ingenuity: An Engineer Looks At Technology And Culture*, ISBN 0-19-513583-0, 2000, Oxford University Press, New York, 2000.

(Luqi 1989) Luqi, "Software Evolution Through Rapid Prototyping", *IEEE Computer*, V22, No. 5 pp 13-25, May 1989

(Luqi 1991) Luqi, "The Role of Prototyping Languages in CASE", *International Journal of Software and Knowledge Engineering*, Vol. 1, No. 2. Pp 131-149, June 1991

(Mandelbrot 1953) Mandelbrot, Benoit, "An Informational Theory of the Statistical Structure of Language," in *Communication Theory*, (ed. By Willis Jackson), Butterworths, 1953.

(Mazur 1978) Mazur, J. & Hastie, R., "Learning As Accumulation: A Reexamination of the Learning Curve," *Psychological Bulletin*, 85 (6), pp. 1256-1274, 1978.

(McCauley 1993) McCauley, Joseph L., *Chaos, Dynamics, And Fractals, An Algorithmic Approach To Deterministic Chaos*, Cambridge University Press, 1993.

(Miller 1956)   Miller G.A. "The Magic Number Seven Plus Or Minus, Two: Some Limits On Our Capacity For Processing Information", *Psychological Review*, 63, pp. 81-97, 1956.

(Misra 1978)   Misra, B., "Nonequilibrium Entropy, Lyapounov Variables, And Ergodic Properties Of Classical Systems," Proceedings, National Academy of Sciences, USA, Physics, Vol. 75, No.4, pp. 1627-1631, April 1978.

MIL-STD-498 Software Development Process Standard, Department of Defense

(Moore 1987)   Moore, G. C., "End-User Computing and Office Automation: A Diffusion of Innovations Perspective," *Infor* 25, pp. 214-235, 1987.

(Moore 1991)   Moore, G., *Crossing the Chasm: Marketing And Selling Technology Products To Mainstream Customers,* Harper Business, 1991.

(Munster 1969)   Munster, Arnold, *Statistical Thermodynamics*, Vol. I, Springer-Verlag, Berlin, 1969.

(Munster 1974)   Munster, Arnold, *Statistical Thermodynamics*, Vol. II, Springer-Verlag, Berlin, 1974.

(Myrtveit 2001)   Myrtveit, Ingunn, Erik Stensrud, Ulf H. Olsson, "Analyzing Data Sets with Missing Data: An Empirical Evaluation of Imputation Methods and Likelihood-Based Methods", *IEEE Transactions of Software Engineering* In the real world there will also be gaps in data.27, No. 11 November 2001.

(Nash 1974)   Nash, L.K., *Elements of Statistical Thermodynamics*, Addison-Wesley Publishing Company, Reading, MA, 1974.

(Nembhard 2000)   Nembhard, David A., "An Individual-Based Description of Learning within an Organization," in *IEEE Transactions on Engineering Management*, Vol 47, No. 3, August, 2000.

(Newell 1981)   Newell, Allen and Rosenbloom, Paul S., "Mechanisms of Skill Acquisition and the Law of Practice," in Anderson, John R., ed., *Cognitive Skills and Their Acquisition*, Lawrence Erlbaum Associates, Hillsdale, NJ, 1981.

(Newton 1726), Newton, Isaac, The Principia, Mathematical Principles of Natural Philosophy, translated by I. Bernard Cohen and Anne Whitman, University of California Press, Berkeley, 1999.

(Nishiyama 2000) Nishiyama, Tetsuto; Ikeda, Kunihhiko; Niwa, Toru, "Technology Transfer Macro-Process, A Practical Guide for the Effective Introduction of Technology," *International Conference of Software Engineering*, Limerick, Ireland, ACM, pp. 577-586, 2000.

(O'Marchu 1997) O'Marchu, Diarmuid, *Quantum Theology*, Crossroads Publishing, New York, 1997.

(Osterweil 1987) Osterweil, Leon, "Software Processes are Software Too," Proceedings, 9th *International Conference on Software Engineering*, pp. 2-13, 1987.

(Pegels 1969) Pegels, C.C., "On Startup of Learning Curves: An Expanded View," *AIIE Trans.*, vol 1. No. 3 pp. 216-222, 1969.

(Pennings 1987) Pennings, J.M., "Technological Innovations in Manufacturing," in *New Technology as Organizational Change*, eds. J.M. Pennings and A. Buitendam,, pp. 197-216, Ballinger., Cambridge, Massachusetts, 1987.

(Peskin 1995) Peskin, Michael E., *An Introduction to Quantum Field Theory*, Addison-Wesley Publishing Company, Reading, MA, 1995.

(Pfleeger 1999) Pfleeger, S.L., "Understanding and Improving Technology Transfer in Software Engineering," *The Journal of Systems and Software* 47, pp. 111-124, 1999.

(Piaget 1963) Piaget, J., *The Origins of Intelligence in Children*, WW Norton, New York, 1963.

(Piaget 1967) Piaget, J., *Six Psychological Studies*, Random House, New York, 1967.

(Piaget 1977) Piaget, J., *The Development of Thought: Equilibration of Cognitive Structures*, Viking Press, New York, 1977.

(Planes 2002)   Planes, Antoni and Vives, Eduard, "Entropic Formulation of Statistical Methods," in Journal of Statistical Physics, Vol. 106, Nos. 314, February 2002.

(Plato c428-c348 B.C.) Plato, "Meno," in the *Dialogues of Plato*, translated by B. Jowett, *Great Books of the Western World*, Book 7, M. Adler, Associate Editor, *Encyclopaedia Britannica, Inc*, ISBN 0-85229, Chicago, pp. 163-9, 1952.

(Poincare 1903)   Poincare, Henri, "Science and Hypothesis," in Physical Sciences, *Great Books of the Western World*, Book 56, M. Adler, Editor in Chief, *Encyclopaedia Britannica, Inc*, Chicago, pp. 1-76, 1952.

(Polanyi 1969)   Polanyi, Michael, *Knowing and Being,* University of Chicago Press, Chicago, 1969.

(Potter 2000) Potter, Marshall, Informal communication with Marshall Potter of the FAA, September, 2000.

(Prigogine 1980) Prigogine, Illya, *From Being to Becoming*, San Francisco, W. H. Freeman, 1980.

(Prigogine 1983)   Prigogine, Illya, "The Rediscovery of Time," a Discourse Prepared for the Isthmus Institute, presented to the American Academy of Religion, December, 1983.

(Prigogine 1984)   Prigogine, Illya, *Order Out of Chaos*, Bantam Books, New York, 1984.

(Prigogine 1989)   Prigogine, Illya, *Exploring Complexity*, *An Introduction*, R.Piper GmbH & Co. KG Verlag, Munich, 1989.

(Przybylinski 1988) Przybylinski, S.M., *Technology Transition: An Annotated Bibliography,*" ACM, Software Eng. Inst., Carnegie Mellon Univ., Pittsburgh, PA, 1988.

(Raghavan 1988) Raghavan, Sridhar, "Diffusion Software Engineering Innovation," THO218-8/88 *IEEE*, pp. 116-118, 1988.

(Raghavan 1989) Raghavan, S.A.; and Chand, D.R. "Diffusing Software-Engineering Methods*," IEEE Software Vol 6, no. 4,* July, 1989.

(Redwine 1984)   Redwine JR, Samuel T., et al, *DoD Related Software Technology Requirements, Practices, and Prospects for the Future*, IDA Paper P-1788, 1984.

(Rogers 1975)   Rogers, Everett M., "Where We Are in Understanding Innovation", paper for the *East-West Communication Conference on Communication and Change: Ten Years Later*, 1975.

(Rogers 1981) Rogers, Everett M., and D. Lawrence Kincaid, *Communication Networks: Toward a New Paradigm for Research,* Free Press, New York, 1981.

(Rogers 1983) Rogers, E. M., *The Diffusion of Innovation*, 3rd ed., Free Press, New York, 1983.

(Rogers 1995) Rogers, E. M., *The Diffusion of Innovation*, 14th ed., Free Press, New York, 1983, 1995.

(Saboe 1989)   Saboe, Michael S., "Software Advanced Technology Transfer (SWATT)," U.S. Army Life Cycle Software Engineering Center Director Meeting, Orlando, FL. September, 1989.

(Saboe 1990)   Saboe, Michael S., "Software Thermodynamics," Joint Logistics Commanders Computer Resource Management Group, December 1989.

(Saboe 1995)   Saboe, Michael S., "Army Software Technology Investment Strategy, Policy Report, HQ Army Material Command, DCSRDA, February, 1995.

(Saboe 2001)   Saboe, Michael S., "A Software Technology Transition Model", Monterey Workshop 2001, Monterey CA, June 2001

(Saboe 2001a) Saboe, Michael S., Grumbacher, P., Kloska, P., "Experience with the WinWin Process for Planning Software Infrastructure and Technology", to be published Int'l Conference on Group Systems Hawaii, 2002.

(Saboe 2001b) Saboe, M., "Technology Transfer: An Annotated Bibliography," Software Engineering Institute Web Page (to be posted), 2002.

(Savage 1954)   Savage, Leonard J., *The Foundations of Statistics*, Wiley, New York, 1954.

(Schroeder 2000)   Schroeder, Daniel V., *Thermal Physics*, Addison Wesley Longman, San Francisco, CA, 2000.

 (Schum 1994)   Schum, David A., *Evidential Foundations of Probabilistic Reasoning, Wiley Series in Systems Engineering*, John Wiley, New York, 1994.

(Shannon 1948)   Shannon, Claude, *A Mathematical Theory of Communication*, Bell Labs, 1948. (available on the internet)

(Shaw 2001)   Shaw, Mary, "Coming-of-Age of Software Architecture Research," *Keynote Address at the 23rd International Conference on Software Engineering,* www.cs.cmu.edu/~shaw/, May 2001

(Simon 1955)   Simon, Herbert A., "On A Class of Skew Distribution Functions," *Biometrika* 42, pp. 425-440, 1955.

(Snoddy 1926)   Snoddy, G.S., "Learning and Stability*,*" *Journal of Applied Psychology*, 10, pp. 1-36, 1926.

(Thurstone 1919)   Thurstone, L.L., "The Learning Curve Equation," *Psychological Monographs*, 26 114), 51, 1919.

Transferring Software Engineering Tool Technology, Conference Paper (Cat. No. 88TH0218-8), Santa Barbara, CA., Nov. 15-16, 1987, pp. vii-165, IEEE Computer Soc. Press, Washington, DC, 1987|.

(Uspensky 1992)   Uspensky, V.A. , "Complexity and Entropy: An Introduction to the Theory of Kolmogorov Complexity," in O. Watanabe, editor, *Kolmogorov, Complexity and Computational Complexity*, pp. 85-102, Springer-Verlag, 1992.

(Uspensky 1992a)   Uspensky, V.A., "Kolmogorov and Mathematical Logic," *Journal of Symbolic Logic*, 57 (2): pp. 385-412, 1992.

(Van de Ven 1991) Van de Ven, A.H., "Managing the process of Organizational Innovation" in *Changing and Redesigning Organizations*, ed. G.P. Huber, Oxford University Press, New York, 1991.

(Vigil 1994) Vigil, and Sarper, H. "Estimating the Effects of Parameter Variability on Learning Curve Model Predictions," *Int. J. Prod. Econ.*, vol. 34, pp. 187-200, 1994.

(von Neumann 1944) von Neumann, J & Morgenstern, *Theory of Games and Economic Behaviour,* 1944.

(Watts 2000) Watts, Robert, "Knowledge Discovery Using the Tech OASIS; Meeting the Information Infrastructure Needs," in proceedings of the Portland International Conference on Management of Engineering and Technology, July 29 - Aug. 2, 2000

(Wehrl 1978) Wehrl, Alfred, "General Properties of Entropy," Reviews of *Modern Physics*, Vol 50, No. 2, April 1978.

(Whitehead 1910) Whitehead, A.N. and Russell, B., *Principia Mathematica*, Cambridge University Press, London, 1910.

(Wiio 1980) Wiio, Osmo A., *Information and Communication: A Conceptual Analysis*, Helsinki, Finland, University of Helsinki, Department of Communications Report, 1980.

(Wright 1936) Wright, T.P., "Factors Affecting the Cost of Airplanes," *Journal of Aeronautical Science*, vol. 3, no. 2 pp. 122-128, 1936.

(Yelle 1979) Yelle, L.E. "The Learning Curve: Historical Review and Comprehensive Survey," *Decision Sci.*, vol. 10, no 2, pp. 302-328, 1970.

(Zelkowitz 1995) (Zelkowitz, Marvin V., "Assessing Software Engineering Technology Transfer Within NASA," NASA Technical Report NASA-RPT-003095, National Aeronautics and Space Administration, Washington, DC, January 1995.

(Zelkowitz 1998) Zelkowitz, Marvin V., Dolores R. Wallace and David Binkley, *Understanding The Culture Clash In Software Engineering Technology Transfer*, University of Maryland Technical Report, 2 June 1998.

(Zipf 1949) Zipf, George Kingsley, *Human Behavior and the Principle of Least Effort,* Addison-Wesley Press, 1949.

(Zipf 1965) Zipf, George Kingsley, *Human Behavior and the Principle of Least Effort and Introduction to Human Ecology*, Hafner Publishing, New York, 1965, (facsimile of the 1949 edition).

(Zurek 1989) Zurek, W.K. (ed) *Complexity, Entropy And The Physics Of Information,* Addison-Wesley, pp. 73-89, 1989.

BIBLIOGRAPHY

PHILOSOPHY REFERENCES

(Aiken 1956)   Aiken, Henry D., *The Age of Ideology*, New American Library, New York, 1956.

(Alexander 1964,1971) Alexander, Christopher, *Notes on the Synthesis of Form*, Harvard University Press, Cambridge, MA, 1964, 1971.

(Bruner, 1956) Bruner, J.S., J.J. Goodnow, and G.A. Austin, *A Study of Thinking*, John Wiley & Sons, New York, 1956.

(Bush 1945) Bush, Vannevar, "As We May Think," *Atlantic Monthly*, (176,1), pp. 101-108, 1945.

(Bush 1947) Bush, Vannevar, *Endless Horizons*, New York, 1947.

(Bucciarelli 1988) Bucciarelli, Louis L., "Engineering Design Process," in Frank A. Dubinskas, ed., *Making Time. Ethnographies of High-Technology Organizations*, Temple University Press, Philadelphia, pp. 92-122, 1988.

(Descartes 1637) Descartes, Rene, *Discourse on the Method*, *Great Books of the Western World* Book 31, M. Adler, Associate Editor, Encyclopaedia Britannica, Inc, ISBN 0-85229, Chicago, pp. 41-67, 1952

(Edman 1944) Edman, Irwin, *Epictetus – Discourses and Enchiridion*, J. Walter Black, Inc., Roslyn, New York, 1944.

(Haskins 1927) Haskins, C., *The Renaissance of the Twelfth Century,* Harvard, 1927.

(Lienhard 2000) Leinhard, John H., *The Engines of Our Ingenuity: An Engineer Looks At Technology And Culture,* ISBN 0-19-513583-0, 2000, Oxford University Press, New York, 2000.

(Plato c428-c348 B.C.) Plato, "Meno," in the *Dialogues of Plato,* translated by B. Jowett, *Great Books of the Western World*, Book 7, M. Adler, Associate Editor, Encyclopaedia Britannica, Inc, ISBN 0-85229, Chicago, pp. 163-9, 1952.

(Puget 1993) Puget, John F., and Christopher K. Ansell, "Robust Action and The Rise of The Medici," 1400-1434, *American Journal of Sociology*, 98: pp.1259-1319, 1993.

(Robinson 1995) Robinson, Daniel N., *An Intellectual History of Psychology*, The University of Wisconsin Press, 1995.

(Schon 1983) Schon, Donald A., *The Reflective Practitioner: How Professionals Think in Action*, Basic Books, New York, 1983.

(Simon 1957) Simon, Herbert A., *Models of Man: Social and Rational*, John Wiley & Sons, New York, 1957.

(Suppe 1974) Suppe, Frederick, "The Search for Philosophic Understanding of Scientific Theories", in F. Suppe, ed., *The Structure of Scientific Theories*, pp. 3-241, University of Chicago Press, Urbana, IL, 1974.

PSYCHOLOGY THEORY REFERENCES

(Anderson 1980) J. R. Anderson, *Cognitive Psychology and Its Implications*, New York, 1980.

(Boring 1930) Boring, E.G., "A New Ambiguous Figure," *American Journal of Psychology*, 42: pp. 444-445, 1930.

(Bruce 1985) Bruce, Darryl, "The How and Why of Ecological Memory," *Journal of Experimental Psychology*, General, 114:, pp. 78-90, 1985.

(Bruner 1986) Bruner, Jerome, *Actual Minds, Possible Worlds*, Harvard University Press, Cambridge, MA, 1986.

(Flexser and Tulving 1982) J. Flexser and E. Tulving, "Priming and Recognition Failure," *Journal of Verbal Learning and Verbal Behavior,* Vol. XXI (1982), pp. 237-48, 1982.

(Glass 1992) Glass, Robert L., Iris Vessey, and Sue A. Conger , "Software Tasks, Intellectual or Social?," *Information & Management*, 23: pp.183-191, 1992.

(Heylighen 1992) Heylighen F., "Principles of Systems and Cybernetics: An Evolutionary Perspective", in *Cybernetics and Systems '92*, R. Trappl (ed.), World Science, Singapore, pp. 3-10. http://pcp.lanl.gov/SELVAR.html, 1992.

(James 1890) James, William, "The Principles of Psychology," *Great Books of the Western World,* Book 53, M. Adler, Associate Editor, *Encyclopaedia Britannica*, *Inc.*, ISBN 0-85229, pp. 163-9, Chicago, 1952.

(Olton 1976) Olton, Robert M. and David M. Johnson, "Mechanisms of Incubation in Creative Problem Solving," *American Journal of Psychology*, 89: pp. 617-630, 1976.

(Olton 1979) Olton, Robert M., "Experimental Studies of Incubation: Searching for the Elusive", *Journal of Creative Behavior*, 13: pp. 9-22, 1979.

(Skinner 1938) B.F. Skinner, *The Behavior of Organisms,* New York, 1938.

(Skinner 1958) B. F. Skinner, "The Science of Learning and the Art of Teaching," *Harvard Educational Review,* Vol. XXIV, pp. 86-97, 1958.

(Skinner 1971) B.F. Skinner, *Beyond Freedom And Dignity,* New York, 1971.

(Solomond 1980) R. Solomond, "Opponent-Process Theory of Acquired Motivation," *American Psychologist,* vol. XXXV , pp. 691-712, 1980.

(Wittgenstein 1953) Wittgenstein, Ludwig, *Philosophical Investigations*, Macmillan Publishing, New York, 1953.

(Woodworth 1938) Woodworth, R.S., *Experimental Psychology*, Holt, New York, 1938.

MATH, PHYSICS STATISTICAL MECHANICS, AND THERMODYNAMICS REFERENCES

(Baker 1990) Baker, Gregory L. and Gollub, Jerry B., *Chaotic Dynamics and Introduction,* Cambridge University Press, 1990.

(Bennett 1982) Bennett, Charles H., "The Thermodynamics of Computation – a Review," *International Journal of Theoretical Physics*, Vol. 21, No. 12, 1982.

(Brynjolfsson 1996) Brynjolfsson, Erik, "Information Technology and Productivity: A Review of the Literature," *Advances in Computers*, Academic Press, Vol 43, pp. 179-214, 1996.

(Brynjolfsson 1993) Brynjolfsson, Erik, "Paradox Lost? Firm-level Evidence of High   Returns   to   Information   Systems   Spending," http://ccs.mit.edu/papers/CCSWP162/CCSWP162.html , 1993.

(Çengel 1989) Çengel, Yunus, and Boles, Michael, *Thermodynamics: An Engineering Approach,* ISBN 0-07-010356-9, McGraw-Hill, 1989.

(Condon 1967)   Condon, E.U. and Odishaw, Hough (editors), *Handbook of Physics*, LCCN 66-20002, McGraw-Hill, 1967.

(Condon 1967a) Condon, E.U. Part 5, Chapter 1, "Heat and Thermodynamics," *Principles of Thermodynamics, Handbook of Physics*, LCCN 66-20002, McGraw-Hill, 1967.

(Cox 1993) Cox, D.R.,, Hinkely, D.V., Reid, N., Rubin, D.B. and Silverman, B.W., editors, *Monographs on Statistics and Applied Probability* (Series), Chapman & Hall, London, 1993.

(Dennery 1972) Dennery, Philippe, *An Introduction to Statistical Mechanics*, George Allen & Unwin Ltd., London, 1972.

(Dewan 1997) "The Substitution of Information Technology for Other Factors of Production: A Firm Level Analysis," *Management Science*, Vol. 43, No. 12, December 1997.

(Dosi 1984) Dosi, Giovanni, *Technical Change and Industrial Transformation*, St. Martin's Press, New York, 1984.

(Dosi 1989) Dosi, Giovanni, *Technical Change and Economic Theory*, Pinter Publishers, London, 1989 (?)

(Ellis 1985) Ellis, Richard S., *Entropy, Large Deviations, and Statistical Mechanics,* Springer-Verlag, New York, 1985.

(Faires 1947) Faires, Virgil Moring, *Elementary Thermodynamics*, The Macmillan Company, New York, 1947.

(Fowler 1936) Fowler, R. H., *Statistical Mechanics*, Cambridge University Press, New York and London, 1936.

(Fowles 1962) Fowles, Grant R*., Analytical Mechanics*, Holt, Rinehart and Winston, New York, 1962.

(Froehling 1981) Froehling, Harold, Crutchfield, J.P., Farmer, Doyne, Packard, N.H. and Shaw, Rob, "On Determining the Dimension of Chaotic Flows," *Physica* 3D, pp. 605-617, North-Holland Publishing Company, 1981.

(Gellert 1975) Gellert W., S. Gottwald, M. Hellwich, H. Kastner, and H. Kustner, eds., *The VNR Concise Encyclopedia of Mathematics* , New York, 1975.

(Gershenfeld 1996) Gershenfeld, N., "Signal Entropy and the Thermodynamics of Computation," *IBM Systems Journal*, Vol. 35, Nos.38.4, 1996.

(Gibbs 1902) Gibbs, J. W. *Elementary Principles of Statistical Mechanics,* Yale University Press, New Haven, 1902.

(Glaser 1967) Glaser, Barney G., and Anselm L. Strauss, *The Discovery of Grounded Theory: Strategies for Qualitative Research,* Aldine, New York, 1967.

(Graham 1980) Graham, Alan K. "A Long-Wave Hypothesis of Innovation," *Technological Forecasting and Social Change* 17, pp. 283-311, 1980.

(Gribbin 1984) Gribbin, John, *In Search of Schrodinger's Cat*, *Quantum Physics and Reality*, Bantam Books, Toronto, 1984.

(Hawking 1988) Hawking, Stephen W., *A Brief History of Time, From the Big Bang to Black Holes*, Bantam Books, Toronto, 1988.

(Henderson 2000), "Untangling the Origins of Competitive Advantage," SMJ Special Issue on the Evolution of Firm Capabilities, forthcoming in the *Strategic Management Journal*, 2000.

(Howell 1971) Howell, James E., and Teichroew, Daniel, *Mathematical Analysis for Business Decisions,* Richard D. Irwin, Inc., 1971.

(Huberman 1979) "Chaotic States of Anharmonic Systems in Periodic Fields*,"* *Physical Review Letters*, Vol. 43, No. 23, December 3, 1979.

(Jaynes 1957) Jaynes, E.T., "Information Theory and Statistical Mechanics," *Physical Review*, Vol. 106, Number 4: pp.620-630, May 15, 1957.

(Jaynes 1957a) Jaynes, E.T., "Information Theory and Statistical Mechanics II," *Physical Review*, Vol. 108, Number 2: pp. 171-190, October 15, 1957.

(Katok 1995) Katok, Aanatole, and Hasselblatt, Boris, *Introduction to the Modern Theory of Dynamical Systems,* Press Syndicate of the University of Cambridge, New York 1995.

(Kolmogorov 1965) Kolmogorov, A.N., "Three Approaches to the Quantitative Definition of Information," *Problemy Peredachi Informatsii*, Vol. 1, No. 1, pp. 3-11, 1965.

(Korn 1961) Korn, G.A. and T.M. Korn, *Mathematical Handbook for Scientists and Engineers*, (pp.18.4.12), McGraw-Hill, New York, 1961.

(Lee 1954) Lee, J.F., *Theory and Design of Steam and Gas Turbine Engines,* McGraw-Hill, New York, 1954.

(Levin 1976) Levin, L.A., "Various Measures of Complexity For Finite Objects (Axiomatic Description)" *Soviet Math. Dokl.*, Vol. 17, No. 2, 1976.

(Montroll 1967a)   Montroll, E.W., Part 5, Chapter 2, "Heat and Thermodynamics," *Principles of Statistical Mechanics and Kinetic Theory of Gases, Handbook of Physics*, LCCN 66-20002, McGraw-Hill, 1967.

(Mueth 1998)   Mueth, Daniel M., Jaeger, Heinrich M., and Nagel, Sidney R., "Force Distribution in a Granular Medium," *Physical Review E*, Vol, 57, No. 3, March 1998.

(Nelson 1964)   Nelson, Alfred L., Folley, Karl W., Coral, Max, *Differential Equations*, D.C. Heath and Company, Boston, 1964.

(Nidditch 1957) Nidditch, M.A., *Introductory Formal Logic of Mathematics*, The Free Press of Glencoe, Illinois, 1957.

(Nidditch 1960) Nidditch, M.A., *Elementary Logic of Science and Mathematics*, The Free Press of Glencoe, Illinois, 1960.

(Noyes 1967a)   Noyes, Richard M., Part 5, Chapter 9, "Heat and Thermodynamics*" Chemical Kinetics, Handbook of Physics*, LCCN 66-20002, McGraw-Hill, 1967.

(Pinson 2000)   Pinson, Gerard, "Classification of Entropies," Kolmogorov Complexity Homepage, www.cmpa.polytechnique.fr/~bousquet/Kolmogorov/entropies.html, 2000.

(Randles – received for peer review, 11/01) Randles, Theodore J., "Knowledge Combustion: Guidelines for the Dissipation of Information," Department of Computer and Information Science, Cleveland State University, Cleveland, OH.

(Russell 1903)   Russell, Bertrand, *Principles of Mathematics*, Cambridge University Press, Cambridge, 1903.

(Schrodinger 1952) Schrodinger, Erwin, *Statistical Thermodynamics*, University Press, Cambridge, 1952

(Shannon 1948)   Shannon, Claude, *A Mathematical Theory of Communication,* Bell Labs, 1948.

(Shapiro 1953)   Shapiro, A. H., *The Dynamics and Thermodynamics of Compressible Fluid Flow,* Ronald Press Company, New York, 1953.

(Simon 1955) Simon, Herbert A., "On a Class of Skew Distribution Functions," *Biometrica* 42, 425-440, 1955

(Speigel 1998)   Speigel, Murray, *Advanced Mathematics for Scientists and Engineers*, *Schaum's Outline Series*, McGraw Hill, 1998.

(Tabor 1989) Tabor, Michael, *Chaos and Integrability in Nonlinear Dynamics, An Introduction,* John Wiley & Sons, New York, 1989.

(Tolman 1938)   Tolman, R.C., *Principles of Statistical Mechanics*, Oxford University Press, New York and London, 1938.

(Uspensky   ) "Complexity and Entropy: An Introduction to the Theory of Kolmogorov Complexity,"

(Waldrop 1992)   Waldrop, M. Mitchell, *Complexity*, Simon & Schuster, New York, 1992 (?)

(Watson 1968) Watson, James D., *The Double Helix, A Personal Account of the Discovery of the Structure of DNA*, Simon and Schuster, New York, 1968.

(Wehrl 1978)   Wehrl, Alfred, "General Properties of Entropy," *Reviews of Modern Physics,* Vol. 50, No. 2, April 1978.

(Whitehead 1910)   Whitehead, A.N. and Russell, B., *Principia Mathematica*, Cambridge University Press, London, 1910.

(Wang 2001)   Wang, Le Yi, and Lin, Lin, "Information-Based Complexity of Uncertainty Sets in Feedback Control," *IEE Transactions on Automatic Control*, Vol 46, No. 4, April 2001.

(Zvonkin 1970)   Zvonkin, A.K. and Levin, L.A., "The Complexity of Finite Objects and the Development of the Concepts of Information and Randomness by Means of the Theory of Algorithms," 1970.

MANAGEMENT AND ECONOMIC REFERENCES

(Alter 2000)   Alter, Allan E., "Knowledge Management's 'Theory-doing Gap'," *Computerworld,* April 10, 2000.

(   )"Agenda Setting In Organizational Behavior: A Theory Focused Approach, *"Journal of Management Inquiry,* 1: pp. 171-182, 1992.

(Astley 1984)   Astley, W. Graham, "Subjectivity, Sophistry and Symbolism in Management Science," *Journal of Management Studies*, 21, pp. 259-272, 1984.

(Astley 1985)   Astley, W. Graham, "Administrative Science as Socially Constructed Truth," *Administrative Science Quarterly*, 30:497-513, 1985.

(Barber 1952)   Barber, Bernard, *Science and the Social Order*, Free Press, Glencoe, IL, 1952.

(Barnes 1982) Barnes, Barry, "A Science-Technology Relationship: A Model and a Query," *Social Studies of Science*, 12: pp.166-72, 1982 .

(Brynjolfsson 1994)   Brynjolfsson, Erik and Hitt, Lorin, *Paradox Lost? Firm-level Evidence of High Returns to Information Systems Spending*, MIT Sloan School, Cambridge, Massachusetts, 1994.

(Brynjolfsson 1996)   Brynjolfsson, Erik and Yang, Shinkyu, "Information Technology and Productivity: A Review of the Literature," in *Advances in Computers*, Academic Press, Vol. 43, pp. 179-214, Cambridge, Massachusetts, 1996.

(Buchanan 1969) J.M. Buchanan, *Cost and Choice* , Chicago, 1969.

(Burt 1992a)   Burt, Ronald S., "The Social Structure Of Competition." in N. Nohria and R. G. Eccles (eds.), *Networks and Organizations: Structure, Form, and Action*: 57-91. Boston: Harvard Business School Press, 1992.

(Burt 1992b)   Burt, Ronald S., *Structural Holes: The Social Structure of Competition*. Cambridge, MA: Harvard University Press, 1992

(Cameron 1999)   Cameron, Kim S. and Quinn, Robert E., *Diagnosing and Changing Organizational Culture*, Addison-Wesley, Reading, MA., 1999.

(Cohen 1994) Cohen, Wesley M., and Daniel A. Levinthal , "Fortune Favors The Prepared Firm", *Management Science*, 40: pp. 227-251, 1994.

(Condon 1967) Condon, E.U. and Odishaw, Hugh, editors, *Handbook of Physics, Second Edition*, McGraw-Hill, New York, 1967.

(Cyert 1963) Cyert, Richard M., and James G. March, *A Behavioral Theory of the Firm*, Englewood Cliffs, NJ: Prentice-Hall, 1993.

(DeMarco 1999)   DeMarco, Tom, and Lister, Timothy, *Peopleware, Productive Projects and Teams*, Second Edition, Dorset House Publishing Co., New York, 1999.

(DiMaggio 1992) DiMaggio, Paul, "Nadel's Paradox Revisited: Relational And Cultural Aspects Of Organizational Structure." in N. Nohria and R. G. Eccles (eds.), *Networks and Organizations: Structure, Form, and Action*, pp.118-142, Boston: Harvard Business School Press, 1994.

(Fernandez 1994)   Fernandez, Roberto M., and Roger V. Gould, 1994 "A Dilemma Of State Power: Brokerage And Influence In The National Health Policy Domain," *American Journal of Sociology*, 99: pp. 1455-1491, 1994.

(Huber 1991)   Huber, George P., "Organizational Learning: The Contributing Processes and the Literatures," *Organizational Science*, Vol. 2, NO. 1, February 1991.

(Ilan 1983) Ilan, Yael, "Evaluation Model in a Dynamic Environment: The Case of the Experience Curve," Stanford University, UMI Dissertation Services, 1983.

(March 1958) March, James G., and Herbert A. Simon, *Organizations*, Wiley, New York, 1958.

(March 1972) March, James G. 1972 "Model Bias In Social Action," *Review of Educational Research*, 44: pp. 413-429, 1972.

(March 1976) March, James G., and Johan P. Olsen, *Ambiguity and Choice in Organizations,* Bergen, Norway: Universitetsforlaget, 1976.

(Meacham, 1990) Meacham, John A., "The Loss of Wisdom," in R. J. Sternberg (ed.), *The Nature of Creativity*, pp. 81-211. Cambridge University Press, New York, 1990.

(Merton 1973) Merton, Robert K., *The Sociology of Science: Theoretical and Empirical Investigations,* University of Chicago Press, Chicago, 1973.

(Miles 1994) Miles, Matthew B., and A. Michael Huberman, *Qualitative Data Analysis*, Sage, Thousand Oaks, CA, 1994.

(Neustadt 1986) Neustadt, Richard E., and Ernest R. May 1986 *Thinking in Time: The Uses of History for Decision Makers*, Free Press, New York, 1986.

(Ogburn 1922) Ogburn, William F., *Social Change*, B. W. Huebsch, New York, 1922.

(Sahakian 1976) S. Sahakian, *Learning: Systems, Models, and Theories,* 2nd ed. Chicago, 1976.

(Schon 1993) Schon, Donald A., "Generative Metaphor: A Perspective On Problem-Setting In Social Policy," in A. Ortony (ed.), *Metaphor and Thought,* pp.137-163. Cambridge University Press, Cambridge, 1993.

(Schumpeter 1934) Schumpeter, Joseph, *The Theory of Economic Development.* Harvard University Press, Cambridge, MA, 1934.

(Skinner 1986), Skinner, Wickham (1986), "The Productivity Paradox," *Harvard Business Review*, July-August, pp. 55-59, 1986.

(Walsh 1987) Walsh, James P., and Robert D. Dewar, "Formalization and the Organizational Life Cycle," *Journal of Management Studies*, 24: pp. 216-231, 1987.

(Walsh 1991) Walsh, James P., and Gerardo R. Ungson, "Organizational Memory," *Academy of Management Review*, 16: pp. 57-91, 1991.

(Weick 1979a) Weick, Karl E., *The Social Psychology of Organizing*, Addison-Wesley, Reading, MA, 1979.

1979b "Cognitive Processes In Organizations." in B. M. Staw (ed.), *Research in Organizational Behavior*, 1: pp. 41-74. JAI Press, Greenwich, CT, 1979.

1994 *Exploring the Black Box: Technology, Economics, and History*. New York: Cambridge University Press, 1992.

GAME THEORY REFERENCE

(Cournot 1898) Cournot, A, *Recherches sur les Principes Mathematiques de la Theorie des Richesses*, 1898.

(Kreps 1989) Kreps, D. (1989a) *Game Theory and Economic Modeling*, 1989.

(Kreps 1990) Kreps, D., *A Course in Microeconomic Theory*, 1990.

(von Neumann 1944)   von Neumann, J & Morgenstern, *Theory of Games and Economic Behaviour,* 1944.

(Rasmusen 1989) Rasmusen, E., *Games and Information,* 1989.

(Sargent 1992) Sargent, T., Book Review in *The Journal of Political Economy,* 1992.

(Sonnenschein 1989) Sonnenschein, H., "Oligopoly and Game Theory", in *The New Palgrave: Game Theory*, 1989.

(Spence 1974) Spence, M., *Market Signalling*, 1974.

(Tirole 1989) Tirole, J., *The Theory of Industrial Organisation*, 1989.

(Varian 1987) Varian, H., *Microeconomic Analysis*, 1987.

TECHNOLOGY TRANSFER REFERENCES

(Aaen 1994)   Aaen, Ivan, "Problems in CASE Introduction: Experiences From User Organizations," *Information and Software Technology*, 1994.

(Allen 1977)   Allen, Thomas J., *Managing the Flow of Technology,* MIT Press, Cambridge, MA, 1977.

(Allen 1983) Allen, Thomas J., Diane B. Hyman, and David L. Pinckney, "Transferring Technology to the Small Manufacturing Firm: A Study of Technology Transfer in Three Countries," *Research Policy,* 12, pp. 199-211, 1983.

(Ardis 1998)   Ardis, Mark A., and Green, Janel A., "Successful Introduction of Domain Engineering into Software Development," *Bell Labs Technical Journal*, July-September 1998.

(Armour 2000)   Armour, Phillip G., "The Case for a New Business Model," *Communications of the ACM*, Vol.41, NO. 8, August 2000.

(Attewell, 1992) Attewell, Paul 1992 "Technology Diffusion and Organizational Learning: The Case Of Business Computing," *Organization Science*, 3: pp. 1-19, 1992.

(Basalla 1988) Basalla, George, *The Evolution of Technology,* Cambridge University Press, New York, 1988.

(Bayer 1988) Bayer, Judy and Melone, Nancy, "A Framework for Understanding Organizational Adoption of Software Engineering Innovations," *Technology Management Publication* T.M. 1, Inderscience Enterprises Ltd., 1988.

(Bayer 1989) Bayer, Judy and Melone, Nancy, "A Critique of Diffusion Theory as a Managerial Framework for Understanding Adoption of Software Engineering Innovations," *The Journal of Systems and Software* 9, pp. 161-166, 1989.

(Burt 1983)   Burt, Ronald S., "Range." in R. S. Burt and M. J. Minor (eds.), *Applied Network Analysis: A Methodological Introduction,* pp. 176-194. Sage, Beverly Hills, CA, 1983.

(Business Week 1996) Business Week Advertisement for Arthur Andersen and Co., p.85, February 26, 1996.

(Buxton 1991) Buxton, J.N., and Malcolm, R., "Software Technology Transfer," *Software Engineering Journal*, January 1991.

(Callon 1980)   Callon, Michel, "The State and Technical Innovation: A Case Study Of The Electric Vehicle In France," *Research Policy*, 9: pp. 358-376, 1980.

(Clapp 1988)   Clapp, Judith, "Government/Industry Interaction in Ada Software Engineering Tool Technology Transfer," IEEE, THO218-8/88/0000/0067, 1988.

(Clarke 1995)   Clarke, Edmund M., and Wing, Jeannette M., et al., "Formal Methods: State of the Art and Future Directions," in ACM Computing Surveys, Vol. 28, No. 4, December 1996.

(Colyer 2000)   Colyer, Adrian, "From Research to Reward: Challenges in Technology Transfer," ICSE Limerick, Ireland, ACM, 2000.

(Curtis   )   Curtis, Bill, "From MCC to CMM: Technology Transfers Bright and Dim,"

(DeBellis 1995)   DeBellis, Michael and Haapala, Christine, "User-Centric Software Engineering," IEEE 0885-90000/95, 1995.

(Ebenau, 1988) Ebenau, R.G. and Lewski, F.H., "Consultative Training at AT&T Bell Laboratories," IEEE, THO218-8/88/0000/0070, 1988.

(Fields 1988)   Fields, Wendell, "Consulting Consortium and Expert Forums at Hewlett-Packard," IEEE THOO218-8/88, 1988.

(Fowler 1993) Fowler, Pricilla and Levine, L., "Technology Transition Push: A Case Study of Rate Monotonic Analysis (Part 1)," Technical Report, CMU/SEI-93-TR-29, ESC-TR 93-203, December 1993.

(Fowler 1994)   Fowler, Pricilla and Levine, L., *From Theory to Practice: Technology Transition at the SEI,* IEEE Proceedings of the Twenty-Seventh Annual Hawaii International Conference on System Sciences, pp. 483-497, 1994.

(Fowler 1994a) Fowler, Pricilla and Levine, L. "The Challenge of Transferring Software and Information Technology," in B.C. Glasson et al. (ed.) *Business Process Re-Engineering: Information Systems Opportunities and Challenges*, Elsevier Science B.V., North-Holland, 1994.

(Fowler 1995) Fowler, Pricilla and Levine, L., "Technology Transition Pull: A Case Study of Rate Monotonic Analysis (Part 2), Technical Report CMU/SEI –93-204, ESC-TR 93-204, April 1995.

(Freeman 1983) Freeman, P., "Software Engineering: Strategies for Technology Transfer," *Proc. Int'l Computing Symp. Applications Systems Development*, pp.333-351, ACM, Berichte, West Germany, 1983.

(Freeman 1988) Freeman, P., "A Transfer Bridge For Software Technology," *Transferring Software Engineering Tool Technology Conference, Santa Barbara,* Nov. 15-16, 1987|, IEEE, Computer Society Press Washington, D.C., 1988.

(Gibson 1989) Gibson, J.E.; Heilig, V.K., "The Challenge Of Technology Transfer Software Engineering Curriculum," in Fairley, R.; Freeman, P., (ed.), *Issues in Software Engineering Education,* pp. 515-524,Springer-Verlag New York, NY, USA, 1989.

(Glass 1988) Glass, Robert L., "Software Technology Transfer: A Multi-flawed Process," *System Development*, September 1988.

(Gould 1989) Gould, Roger V., and Roberto M. Fernandez, "Structures of Mediation: A Formal Approach To Brokerage In Transaction Networks", *Sociological Methodology*, 19: pp. 89-126, 1989.

(Griss 1994) Griss, M.L., "Software reuse experience at Hewlett-Packard," Proceedings of 16th International Conference on Software Engineering, Sorrento, Italy, 1994.

(Hargadon 1997) Hargadon, Andrew and Sutton, Robert I., "Technology Brokering and Innovation in a Product Development Firm," *Administrative Science Quarterly*, Vol 42, No. 4, December, 1997.

(Hornbach 1988) Hornbach, Katherine, "The Role of Support Staff in the Successful Introduction of New Tool Technology,"" IEEE THO218-8/88/000/0074, 1988.

(Huber 1991) Huber, George P., "Organizational Learning: The Contributing Processes and the Literature." *Organization Science*, 2, pp. 88-115, 1989.

(Hughes 1989) Hughes, Thomas P., *American Genesis: A Century of Invention and Technological Enthusiasm,* Penguin Books, New York, 1989.

(IEEE Std 1348-1995)   IEE Recommended Practice for the Adoption of Computer-Aided Software Engineering (CASE) Tools, ISBN 1-55937-591-4, IEEE, 1996

(Jaakola 1995) Jaakola, Hannu, "Comparison and Analysis of Diffusion Models," p 65-82, 1995.

(James 1995) James, Dilmus D., and Vickers, Danny, "Computer Software in Developing Countries: A Case Study of CD. Juarez, Mexico*," Journal of Global Information Management*, May 8, 1995.

(Jeffrey 1988)   Jeffrey, Joel H. "A Unifying Comprehensive Framework for Software Technology Transfer," IEEE THO218-8/88, pp. 82-85, 1988.

(Kantrow 1987)   Kantrow, Alan M., *The Constraints of Corporate Tradition*. Harper & Row, New York, 1987.

(Katzenbach, 1993) Katzenbach, Jon R., and Douglas K. Smith, *The Wisdom of Teams: Creating the High-performance Organization*, Harvard Business School Press, Boston, 1993.

(Kitson 1993)   Kitson, D.H.; Masters, S.M., "An Analysis Of SEI Software Process Assessment Results: 1987-1991," *Proceedings of 1993 15th International Conference on Software Engineering,* Baltimore, MD, May 17-21, 1993, pp. 68-77, IEEE Computer Society Press, Los Alamitos, CA., 1993.

(Koch 1993) Koch, G.R., "Process Assessment: the 'BOOTSTRAP' Approach," *Information and Software Technology*, Vol. 35, No. 6/7 June/July 1993.

(Krasner 1995)   Krasner, H., "Bottlenecks In The Transfer Of Software Engineering Technology: Lessons Learned From A Consortium Failure," in Nunamaker, J.F., Jr.; Sprague, R.H., Jr.(ed.) *Proceedings of the Twenty-Eighth Annual Hawaii*

International Conference on System Sciences,Wailea, HI, 3-6 Jan. 1995, pp. 635-41 vol.4 IEEE Computer Society Press Los Alamitos, CA,1995.

(Latour 1987)   Latour, Bruno, *Science in Action*, Harvard University Press, Cambridge, MA, 1987.

(Law 1987)   Law, John "Technology And Heterogenous Engineering: The Case of Portuguese Expansion." in W. E. Bijker, T. P. Hughes and T. Pinch (eds.), *The Social Construction of Technological Systems,* pp.111-134. MIT Press, Cambridge, MA, 1987.

(Lieblein 1985)   Lieblein, Edward, "An Overview of the DoD Software Initiative," EASCON '83: 16th Annual IEEE Electronics and Aerospace Systems Conference and Exposition. Proceedings,| Oct. 1983|.

(Linger 1993)   Linger, Richard C., and Hevner, Alan R., "Achieving Software Quality Through Cleanroom Software Engineering," IEEE 0-8186-1060-3425, 1993.

(Linger 1992)   Linger, R.C.; Spangler, R.A., "The IBM Cleanroom Software Engineering Technology Transfer Program," in Sledge, C.| (ed*.), Software Engineering Education. SEI Conference 1992 Proceedings*, San Diego, CA, 5-7 Oct. 1992|, pp. 380-94 Springer-Verlag Berlin, Germany, 1992.

(Lucic 1995)   Lucic, Richard A., and Rohreer, Ronald, "Undergraduate Field Applications Engineers: A Successful Experiment in Electronic Design Automation Technology Transfer," IEEE, 0018-9359/95, 1995.

(Maguire 1999)   Maguire, Eoin, Jackson, Ian and Flynn, Tom, "Process Improvement Experiment Final Report, Verification & Validation Methodology for JAVA Development Environment," European Systems & Software Initiative, Office Integrated Solutions Limited, Version 2, May 30, 1999.

(Manley 1983)   Manley, John H., "Industry Perspective on Stars," IEEE 0730-3157/83/0000/0077, 1983.

(Marsden 1982)   Marsden, Peter V., "Brokerage Behavior In Restricted Exchange Networks," in P. V. Marsden and N. Lin (eds.), *Social Structure and Network Analysis*: pp. 201-218. Sage, Beverly Hills, CA, 1982.

(Millard 1990) Millard, Andre, *Edison and the Business of Innovation*, Johns Hopkins University Press, Baltimore, 1990.

(Mitchell 1996) Mitchell, K.I., "Technology Transfer to and From the Industrial Sector," in Purvis, M., (ed.), *Proceedings 1996 Inti'l Conference Software Engineering Education and Practice*, Dunedin, New Zealand, January 24-27, 1996, IEEE Computer Society Press, 1996.

(Moore 1990) Moore, G. *Crossing the Chasm: Marketing And Selling Technology Products To Mainstream Customers,* Harper-Business, 1990.

(Mueller 1975) Mueller, Willard F., "The Origins Of The Basic Inventions Underlying Du Pont's Major Product And Process Innovations, 1920 To 1950." in R. R. Nelson (ed.), *The Rate and Direction of Inventive Activity: Economic and Social Factors*, pp.323-358, Princeton University Press, Princeton, 1975.

(Nielsen 1995) Nielsen, Jakob, "Getting Usability Used," in Nordby, K.; Helmersen, P.H.; Gilmore, D.J.; Arnesen, S.A. (ed.)(c) 2000 Institution of Electrical Engineers. - *Proceedings of 5th International Conference on Human-Computer Interaction* (INTERACT'95)| Lillehammer, Norway June 27-29,1995, Chapman & Hall London, UK, 1995.

(Osborn 1957) Osborn, Alex F., *Applied Imagination*, Scribner, New York, 1957.

(Petrovski 1992) Petrovski, Henry, *The Evolution of Useful Things*, New York: Knopf., 1992.

(Payne 1997) Payne, Jeffrey E.,"Why Testing Technology Is Not Transferred To Industry: Academics Don't Get It, Vendors Don't Know It, Practitioners Don't Care," IEEE 0-7803-3979, IEEE, 1997.

(Porter 2000) Porter, A.L., Newman, N.C., Watts, R.J., Zhu, D., and Courseault, C., Matching Information Products to Technology Management Processes, *AAAI (American Association for Artificial Intelligence) Workshop on Bringing Knowledge to Business Processes*, Stanford, CA, April, 2000.

(Porter 2000 a) Porter, A.L., Newman, N.C., Watts, R.J., Zhu, D., Courseault, C., Myers, W., and Yglesias, E., Why Don't Technology Managers Want our Knowledge? International Association for Management of Technology, Miami, 2000.

(Porter 1998-2000)   Porter, A.L., Carlisle, J.P., Watts, R.J. *" Mining Bibliographic Information on Emerging Technologies"*, U.S. National Science Foundation, Management of Technological Innovation Project (DMI-9872482), 1998-2000.

(Raghavan 1989) Raghavan, Sridhar and Chand, Donald R., "Diffusing Software-Engineering Methods," 0740-7459/89 IEEE, pp. 81-89, 1989.

(Redwine 1984) Redwine Jr, Samuel T., et al, *DoD Related Software Technology Requirements, Practices, and Prospects for the Future,* IDA Paper P-1788, 1984.

(Ribot 1906) Ribot, T., *Essays on the Creative Imagination*, Routledge & Kegan Paul. London, 1906.

(Rogers 1983) Rogers, E. M., *The Diffusion of Innovation*, 3rd ed., Free Press, New York, 1983.

(Rogers 1995) Rogers, E. M., *The Diffusion of Innovation*, 4th ed., Free Press, New York, 1983.

(Rombach 2000) Rombach, Dieter, "Franhofer: The German Model for Applied Research and Technology Transfer," ICSE, Limerick Ireland, ACM, 2000.

(Rosenberg 1982),   Rosenberg, Nathan, *Inside the Black Box*, Cambridge University Press, New York, 1982.

(Scacchi 1988) – Scacchi, W., "Understanding Software Technology Transfer: Barriers To Innovation Engineering," IEEE, THO218-8/88/0000/0130, 1988.

(Schneider 2000)   Schneider, Thomas, "Information Theory Primer," www. LECBNCIFCRF gov-toms/paper/primer, 2000.

(SEI 2000) www. SEI.cmd.edu/ "Readings in Software Technology Transition", obtained in 2000.

(Sprague 1999)   Sprague, Ralph H., *Proceedings of the 32nd Annual Hawaii International Conference on System Sciences, Abstracts and CD-ROM of Full Papers*, January 5-8, 1999, IEEE Computer Society, Los Alamitos, CA, 1999.

(Staudenmaier 1985)   Staudenmaier, John M., *Technology's Storytellers: Reweaving the Human Fabric*, MIT Press , Cambridge, MA, 1985

(Stewart 1996) Stewart, Thomas A. "3M Fights Back." *Fortune Magazine*, pp. 94-99, February 5, 1996.

(Sutton 1996)   Sutton, Robert I., and Andrew B. Hargadon, "Brainstorming Groups In Context: Effectiveness In A Product Design Firm." *Administrative Science Quarterly*, 41. Pp. 685-718, 1996.

(Torrance 1988) Torrance, E. P., "The Nature Of Creativity As Manifest In Its Testing." in R. J. Sternberg (ed.), *The Nature of Creativity*, pp. 43-75. Cambridge University Press, New York, 1988.

(Usher 1929)   Usher, Abbot Payton, *A History of Mechanical Inventions*, New York: McGraw-Hill, 1929.

(U.C. Berkeley 1997)   "IR Implementation Issues, Web Crawlers, and Web Search Engines," SIMS 202: Information Organization and Retrieval, August 28, 1997.

(Watts 1997)   Watts, R.J., and Porter, A.L., "Innovation Forecasting," *Technological Forecasting and Social Change*, Vol. 56, p. 25-47, 1997.

(Watts 1997a) Watts, R.J., Porter, A.L., Cunningham, S.W., and Zhu, D., "TOAS Intelligence Mining: Analysis of Natural Language Processing and Computational Linguistics," in J. Komorowski and J Zytkow (eds.), *Principles of Data Mining and Knowledge Discovery (First European Symposium, PKDD'97, Trondheim, Norway)*, p. 323-335: Springer, 1997.

(Watts 2000)   Watts, R.J., Courseault, C.R., Kaplin, S.J., "Identifying Unique Information Using Principal Component Decomposition," International Association for Management of Technology, Miami, 2000.

(Watts 1999)   Watts, R.J., Porter, A.L., Courseault, C., "Functional Analysis: Deriving Systems Knowledge from Bibliographic Information Resources," *Information, Knowledge, Systems Management*, 1 (1999) 1-16 IOS Press.

(Watts 1999a)   Watts, R.J.; Porter, A.L., Mining Foreign language Information Resources, *Proceedings, Portland International Conference on Management of Engineering and Technology (PICMET)*, Portland, OR, USA, July, 1999.

(Watts 1998)   Watts, R.J., Porter, A.L., and Newman, N.C., "Innovation Forecasting Using Bibliometrics," *Competitive Intelligence Review*, Vol. 9, No. 4, p. 1-9, 1998.

(Watts 1998a)   Watts, R.J., and Porter, A.L., "Innovation Forecasting using Functional/Capabilities Analyses," International Symposium on Forecasting, Edinburgh, 1998.

(Willis 1983)   Willis, R.R., "Technology Transfer Take 6 +/- 2 Years," IEEE, CHI 883-8/83/0000//01085, 1983.

(Yourdon 1987)   Yourdon, Edward; A Game Plan for Technology Transfer, Yourdon Inc., New York, 1987.

(Zukier 1984) Zukier, Henri and Albert Pepitone, "Social Roles and Strategies in Prediction: Some Determinants of the Use of Base Rate Information," *Journal of Personality and Social Psychology*, (47,2), 1984.

SOFTWARE REFERENCES

(Abrahamsen, 1987)   Abrahamsen, Adele A.,"Bridging Boundaries Versus Breaking Boundaries: Psycholinguistics in Perspective," *Synthese*, 72: pp. 355-388, 1987.

(Adler 1992b) Adler, Paul S. and Terry A. Winograd, eds. (1992b), *Usability. Turning Technology into Tools*, Oxford. New York, 1992.

(Balzer 1983)   Balzer, Robert, Thomas E., Cheatham, Jr., and Cordell Green , "Software Technology in the 1990's: Using a New Paradigm," *Computer* (16,11):pp.39-45, 1983.

(Basili 1991)   Basili, Victor R., and Musa, John D., "The Future Engineering of Software: A Management Perspective," *Computer,* pp. 90-96, September 1991.

(Basili 1994)   Basili, Victor R., Selby, Richard W., and Hutchens, David H., *Experimentation in Software Engineering*, The Institute of Electrical and Electronics Engineers, Inc. 1994.

(Boehm 2000) Boehm, Barry and Basili, Victor R., "Gaining Intellectual Control of Software Development," IEEE, *Computer*, May, 2000.

(Boehm 2000a)   Boehm, Barry W., Abts, Chris, Brown, A. Winsor, Chulani, Sunita, Clark, Bradford K., Horowitz, Ellis, Madachy, Ray, Reifer, Donald J., and Steece, Bert, "Software Cost Estimation with COCOMO II," Prentice Hall PTR, Upper Saddle River, N.J., 2000.

(Boehm 2001)   Boehm, Barry, Basili, Vic, "Achieving CMMI Level 5 Improvements via Experience Factory Practices," *DoD Software Collaborators' Meeting*, February 12, 2001, Center for Empirically-Based Software Engineering, 2001.

(Bernstein 1966)   Bernstein, J., *The Analytical Engine*, Random House, New York, 1966.

(Bloomfield 1992) Bloomfield, Brian P., "Understanding the Social Practices of Systems Developers," *Journal of Information Systems*, 2: pp.189-206, 1992.

(Blum 1989a)   Blum, B.I., "Volume, Distance and Productivity," *Journal of Systems and Software*, 10: pp.217-226, 1989.

(Blum 1989b) Blum, B.I., "Improving Software Maintenance by Learning From the Past: A Case Study," *Proceedings of IEEE*, 77: pp.596-606, 1989.

(Blum 1990a)   Blum, B.I., *TEDIUM and the Software Process*, MIT Press, Cambridge, MA, 1990.

(Blum 1991a)   Blum, B.I., "A Ten-Year Evaluation of an Atypical Software Environment" *Software Engineering Journal*, 6:347-354.

(Blum 1991b)   Blum, B.I., "Towards a Uniform Structured Representation for Application Generation," *International Journal of Software Engineering and Knowledge Engineering*, 1, pp. 39-55, 1991.

(Blum 1991c)   Blum, B.I., "Integration Issues Elucidated In Large-Scale Information System Development," *Journal of Systems Integration*, 1:pp 35-53, 1991.

(Blum 1992b)   Blum, B.I., *Software Engineering: A Holistic View*, Oxford University Press, New York, 1992.

(Blum 1992d)   Blum, B.I., "A Multidimensional Approach to Application Development," *Computer Applications and Design Abstractions*, PD-Vol. 43, ASME Press, pp. 87-90, 1992.

(Blum 1993)   Blum, B.I., "The Economics of Adaptive Design, " *Journal of Systems and Software*, 21:pp.117-128, 1993.

(Blum 1993b)   Blum, B.I., "Representing Open Requirements with a Fragment-Based Specification", *IEEE Transactions on Systems, Man and Cybernetics*, 23:pp. 724-736, 1993.

(Blum 1994b)   Blum, B.I., "Characterizing the Software Process," *Information and Decision Technologies*, 19:pp.215-232, 1994.

(Blum 1996) Blum, B.I., *Beyond Programming*, *To a New Era of Design,* Oxford University Press, New York, 1996.

(Boehm 1981)   Boehm, Barry W., *Software Engineering Economics*, Prentice-Hall, Englewood Cliffs, 1981.

(Boehm 1988) Boehm, Barry W., "A Spiral Model of Software Development and Enhancement," *Computer*, (21,5), pp.61-72, 1988.

(Cox 1990) Cox, Brad J., "Planning the Software Industrial Revolution," *IEEE Software*, pp. 25-33, November, 1990.

(Calabrese 1990)   Calabrese, Philip, "Reasoning with Uncertainty Using Conditional Logic and Probability," in Proceedings of the First International Symposium on Uncertainty Modeling and Analysis, Ayyub, Bilal M., ed., pp. 682-688, IEEE Computer Society Press, 1990.

(Cusumano 1995)   Cusumano, Michael A. and Selby, Richard W. (1995); *Microsoft Secrets: How the World's Most Powerful Software Company Creates Technology, Shapes Markets, and Manages People*; The Free Press/Simon & Schuster, New York, 1995.

(Cusumano 1998)   Cusumano, Michael A. and Yoffie, David B. (1998); *Competing on Internet Time*, Touchstone, New York, 1998.

(Freeman 1989) Freeman, Peter, "Strategic Directions in Software Engineering: Past, Present, and Future," Information Processing 89, G.X. Ritter (ed.), Elsevier Science Publishers B.V., North-Holland, 1989.

(Glass 1994) Glass, Robert L., in the "Year 2020, The Computing and Software Research Crisis," *IEEE Software*, November, 1994.

(Glass 1995) Glass, Robert L., "Research Phases in Software Engineering: An Analysis of What is Needed and What is Missing," *Journal of Systems and Software*, pp.28: pp.3-7, 1995.

(Glass 1998)   Glass, Robert L, "An Assessment of Systems and Software Engineering Scholars and Institutions (1993-1997)," The Journal of Systems and Software 43 pp. 59-64, Elsevier, 1998.

(Grable 1994) Grable, Ross, *"A State Based Software Entropy Metric"*, US Army Missile Command, Personal communication, November 11, 1994.

(Hall 1995)   Hall, Elaine Marie, "Proactive Risk Management Methods for Software Engineering Excellence," Ph.D dissertation, Computer Science, Graduate School of Florida Institute of Technology, Melbourne, Fla, 1995.

(Hall 1998)   Hall, Elaine M., *Managing Risk, Methods for Software Systems Development*, Addison-Wesley, Reading, MA., 1998.

(Humphrey 1989) Humphrey, Watts S., *Managing the Software Process*, Addison-Wesley Publishing Company, Reading, MA, 1989.

(Kulch 1972) Kulch, W. *"Entropy of Transformed Finite State Automata and Associated Languages"*, in *Graph Theory and Computation*, R.C. Read, ed. Academic Press, New York, 1972.

(Lyu 1996) Lyu, Michael R., ed*., Handbook of Software Reliability Engineering*, IEEE Computer Society Press, McGraw-Hill, New York, 1996.

(Osterweil 1987) Osterweil, Leon, "Software Processes are Software Too," *Proceedings, 9th International Conference on Software Engineering*, pp. 2-13, 1987.

(Pham 1995) Pham, Hoang, *Software Reliability and Testing*, IEEE Computer Society Press, Los Alamitos, CA., 1995.

(Potts 1993) Potts, Colin, "Software-Engineering Research Revisited," *IEEE Software*, September: pp. 19-28, 1993.

(Prowell 1999) Prowell, Stacy J., Trammell, Carmen J, Linger, Richard, Poore, Jesse H., *Cleanroom Software Engineering, Technology and Process*, Addison Wesley Longman, Reading, MA., 1999.

(Pukite 1998) Pukite, Jan and Pukite, Paul, *Modeling for Reliability Analysis*, IEEE Press, , New York, 1998.

(Raffo 1997) Raffo, D.M.; Vandeville, J.V., "Quantitative Process Modeling As A Basis For Managing Software Engineering Process Improvement," in Kocaoglu, D.F.; Anderson, T.R. (ed.) - *Innovation in Technology Management. The Key to Global Leadership,* pp. 609-612, Portland International Conference on Management of Engineering Technology, July 27-31, 1997, IEEE , New York, NY, 1997.

(Salzman 1992) Salzman, Harold, "Skill-Based Design: Productivity, Learning and Organizational Effectiveness," in Paul S. Adler and Terry A. Winograd, eds., *Usability. Turning Technology into Tools*, Oxford, New York, pp. 66-96, 1992.

(Shaw 1990) Shaw, Mary "Prospects for an Engineering Discipline of Software," *IEEE Software*, November: pp. 15-24, 1990.

(Shrestha 1999) Shrestha, Jayesh Man, "Modeling Database Applications Using the Unified Modeling Language," University of Texas at Arlington, 1998, UMI Dissertation Services, Ann Arbor, MI., 1999.

(Soloway 1984)   Soloway, Elliot and Kate Ehrlich, "Empirical Studies of Programming Knowledge," *IEEE Transactions on Software Engineering*, 10: pp. 595-609, 1984.

(Staudenmayer 1998)   Staudenmayer, Nancy and Cusumano, Michael, "Alternative Designs for Product Component Integration, working paper, MIT Sloan School of Management, April 1998.

(Tauber 1991)   Tauber, M.J. and D. Ackermann, eds., *Mental Models and Human-Computer Interaction II,*, Amsterdam: North-Holland, 1991.

(Wiio 1980) Wiio, Osmo A., *Information and Communication: A Conceptual Analysis*, Helsinki, Finland, University of Helsinki, Department of Communications Report, 1980.

(Woodcock 1998) Woodcock, Jim and Loomes, Martin, *Software Engineering Mathematics,* Addison -Wesley Publishing Company, Reading, MA., 1998.

SYSTEMS ENGINEERING REFERENCES

(Barnes 1982) Barnes, Barry, "The Science-Technology Relationship – A Model and a Query," *Social Studies of Science* (SAGE, London and Beverly Hills), Vol. 12, 1982.

(Cohen 1994) Cohen, Wesley M. and Levinthal, Daniel A*, Fortune Favors the Prepared Firm,* The Institute of Management Sciences, Vol. 40, No. 2, 1994.

(Easton 1952) Easton, Stewart C., *Roger Bacon and His Search for a Universal Science,* Basil Blackwell, Oxford, 1952.

(Glass 1998)   Glass, Robert L., "An Assessment of Systems and Software Engineering Scholars and Institutions" (1993-1997), *The Journal of Systems and Software: Computing Trends*, Bloomington, IN., 1998.

(Hargadon 1997)   Hargadon, Andrew and Sutton, Robert I., *Technology Brokering and Innovation in a Product Development Firm,* Cornell University, 1997.

(Huber 1991)   Huber, George P., *Organizational Learning: The Contributing Processes and the Literatures*, The Institute of Management Studies, 1991.

(Sage 1990a) Sage, Andrew P., *Concise Encyclopedia of Information Processing in Systems and Organizations*, Pergamon Press, Oxford, 1990.

(Sage 1990b) Sage, Andrew P., "Design and Evaluation of Systems,' in Andrew P. Sage, *Concise Encyclopedia of Information Processing in Systems and Organizations*, Pergamon Press Oxford; 1990.

(Sage 1992) Sage, Andrew P., *Systems Engineering*, John Wiley & Sons, New York, 1992.

(Schuler 1993) Schuler, Douglas and Aki Namioka, eds., *Participatory Design: Principles and Practices*, Lawrence Erlbaum Associates, Hillsdale, NJ, 1993.

(Senders 1991) Senders, John W. and Neville P. Moray, *Human Error: Cause, Prediction, and Reduction*, Lawrence Erlbaum Associates, Hillsdale, NJ, 1991.

THIS PAGE INTENTIONALLY LEFT BLANK

APPENDIX A INFORMATION, CONTROL THEORY AND EVOLUTIONARY DYNAMICAL SYSTEMS BASICS

This appendix reviews some basics of information theory, beyond what is given in Chapter 3. A discussion of the meaning of an operator, the significance of the eigenvalue, and the basics of Markov chains is provided. The content can be found in several common graduate texts, but the references here are readily related to the usage in terms of symbolic dynamics and information used in the technology transfer models developed in this dissertation. There is no attempt made to clarify the development of these topics beyond the very basics to give the reader a quick primer on the subject. All theorems come directly from the reference documents. In those references, there are examples and narrative that can provide a deeper understanding to advance past this appendix primer. The references also provide explicit details on properties and conditions that must be observed for the theorems to hold.

After the presentation of these topics, the reader is provided with a very brief discussion on the relationship of randomness and complexity. Further research can move forward minimizing the burden of taming the mathematical notions using these concepts. This appendix provides some of the deeper mathematical and physics advances reflected in the technology transition models, what is speculated as relevant to evolutionary software development, and software itself. Further discussion on the math, and physics utilized can be found in Prigogine (Prigogine 1983, 1987, 1997), Shannon (Shannon 1948), Jaynes (Jaynes 1957a,b), Kolmogorov (Kolmogorov 1965), Farmer (Farmer 1983), Baker (Baker 1990), and Brown (Brown 1992, 1992a, 1993, 1993a,b,c,d, 1995, 1996, 1996a, 1997, 1998, 1999, 1999a). Of these, the references from Shannon, and Prigogine are the best place to start. Reasonably readable graduate textbooks on information theory and Kolmogorov's complexity are (Cover 1991, or Li 1993) respectively. Baker's text on non linear systems and dynamical fundamentals (Baker 1990) is an easy place to start.

A. INFORMATION THEORY

What follows is a basic review of entropy in information theory after Shannon, Jaynes, Kolmogorov, Uspensky, and others as found in Li, (Li 1993) and Cover (Cover 1991). This review section is drawn from Cover (Cover 1991 p13).

Let X be a discrete random variable with alphabet Ξ and a probability mass function $p(x)=\Pr\{X=x\}$, $x \in \Xi$. $p(x)$ and $p(y)$ refer to two different random variables and are in fact two different probability mass functions $p_x(x)$ and $p_y(y)$. For the alphabet, with the given probability mass function, the definition of information entropy is:

$$S_H(X) = -\sum_{x \in \Xi} p(x) \log_2 p(x) \qquad (A.1)$$

S_H is the entropy measured in bits, and the log is base 2. Log_2 will be assumed throughout unless otherwise noted. For example, the entropy of a fair coin toss is 1 bit. The convention of $0 \log 0 \to 0$ is used, which comes from continuity since $x \log x \to 0^+$, as $x \to 0^+$. Using L'Hôpital's rule $\lim_{x \to 0^+} x = 0$ and $\lim_{x \to 0^+} x = -\infty$ we can convert to the form ∞/∞. (Kreyszig 1993 p500)

The base of the log is two for the natural units of information entropy as developed by Shannon (Shannon 1948). The entropy is a function of the distribution of X. It does not depend on the actual values taken by the random variable X, but only on the probabilities.

If $X \sim p(x)$ which means that the probability of use the random variable is representative of the element's usage over the alphabet, then the expected value E of a random variable $g(X)$ is denoted

$$E_{p(x)} g(X) = \sum_{x \in \Xi} g(x) p(x) \qquad (A.2)$$

The entropy of a plain random variable X can be interpreted as the expected value of $\log \dfrac{1}{p(X)}$, where X is drawn according to the probability mass function $p(x)$. Thus

$$E_{p(x)} \log \frac{1}{p(X)} = \sum \log \frac{1}{p(x)} p(x) = -\sum_{x \in \Xi} \log p(x) p(x) = S_H \qquad \text{(A.3)}$$

1.   Maximum Entropy – Equal Probabilities

Here is an example. Let have a system where there are only two choices.

$$X = \begin{cases} 1 & \text{with probability } p \\ 0 & \text{with probability } 1-p \end{cases} \qquad \text{(A.4)}$$

then

$$S_H(X) = -p \log p - (1-p) \log(1-p) \equiv S_H(p) \qquad \text{(A.5)}$$

We see that $S_H = 1$ bit when $p=1/2$. FigureA-1 shows the basic properties of entropy. It is a concave function of the distribution and equals 0 when $p=0$ or 1. This makes sense because when $p=0$ or 1, the variable is not random and there is no uncertainty. The entropy is maximum when $p=.5$, which corresponds to the maximum value of the entropy.

Entropy vs Probability

Entropy vs. Probability

$Entropy\ S_H$

$$S_H = -\sum p(x)\ log_2\ p(x)$$

$$S_H = -(p)\ log\ p - (1-p)\ log\ (1-p)$$

Expected value

$$E_{p(x)}g(X) = \sum_{x \in \Xi} g(x)p(x)$$

$$E_{p(x)}log\frac{1}{p(X)} = S_H$$
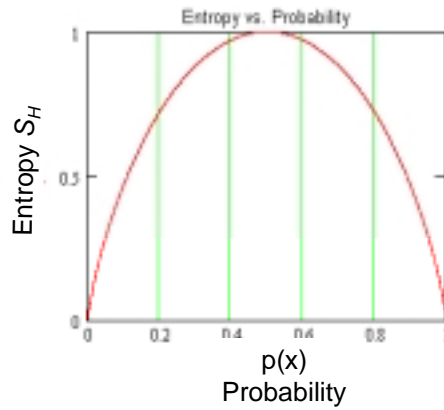
FigureA-1 Entropy vs. Probability

Consider a system where input signals $X \in T$. Specifically, where X is a set of terms,

$$\begin{cases} T \equiv \{term\} \\ 2^T = \{msg\} \end{cases} \qquad (A.6)$$

Where 2^T is a set of all the subsets, often called the power set. Here is an example.

T={A, B, C, D}

$$2^T \equiv \begin{cases} \{\ \},\{A\},\{B\},\{C\},\{D\}, \\ \{A,B\},\{A,C\},... \\ \{A,B,C\},... \\ ...\{A,B,C,D\} \end{cases} \qquad (A.7)$$

Now when the number of elements in |T| =4, we get $2^{|T|} = 2^4 = 16$. The maximum entropy occurs when we have an equal distribution of terms. So for a message set where each subset of terms appears only once we define S_H as

$$S_{H_{max}} = -\sum_{x \in 2^\Xi} \frac{1}{2^{|T|}} \log_2 \frac{1}{2^{|T|}} \tag{A.8}$$

In this example, the maximum entropy $S_{H_{max}} = -\sum_{x \in \Xi}^{2^4} \frac{1}{16} \log_2 \frac{1}{16} = 4$. It is easy to see that the maximum entropy will always be |T|, for the condition that all of the sets of terms in the set are evenly distributed.

$$S_H(|n|) = \sum_{i=1}^{2^n} \frac{1}{|n|} \log_2 \frac{1}{|n|} = n \tag{A.9}$$

$$\{a_1, a_2, ..., a_n\}$$

$$\frac{1}{n}, \frac{1}{n}, ..., \frac{1}{n}$$

$$S_H(p) = -\sum_{i=1}^{2^{\frac{1}{p}}} p \log p = \frac{1}{p} = \frac{1}{|T|} \text{ for } 0 \le p \le 1 \tag{A.10}$$

The entropy maximum is at *1/p(X) or |T|*, or the number of sets of terms in the alphabet T. In Figure A-2 we see the effect of sets of terms that are evenly distributed. In our model, we would *not* expect to see *.5< p(X) <1* as the result of integer number of sets of terms. This is because we make decisions between two choices one set of terms and another set of terms, that yields a probability of .5. If we have one choice, one set of terms, we are certain of the answer, and the probability is 1/1 or by definition S_H=0.

Maximum Entropy

Entropy vs 1/ |T| i.e.or p(X)
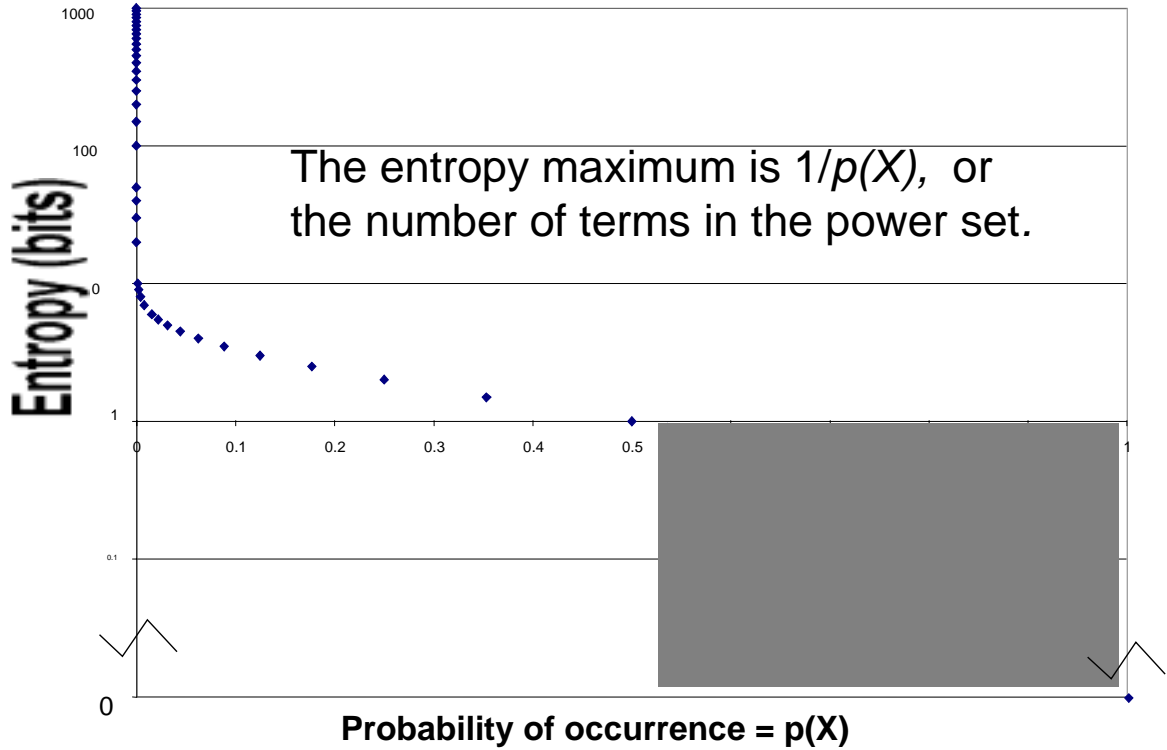
Figure A-2 Even distribution of terms, yields maximum entropy

Here is another example. Let

$$X = \begin{cases} a & \text{with probability } 1/2, \\ b & \text{with probability } 1/4, \\ c & \text{with probability } 1/8, \\ d & \text{with probability } 1/8. \end{cases} \quad \text{(A.11)}$$

The entropy of X is

$$S_H = -\frac{1}{2}\log\frac{1}{2} - \frac{1}{4}\log\frac{1}{4} - \frac{1}{8}\log\frac{1}{8} - \frac{1}{8}\log\frac{1}{8} = \frac{7}{4} \text{ bits (A.12)}$$

Suppose we wish to determine the value of X with the minimum number of binary questions. An efficient first question is "Is $X=a$?" This splits the probability in half. If the answer to the question is no, the second question can be, "Is $X=b$?" The third question is "Is $X=c$?" The resulting expected number of binary questions is 1.75. This turns out to be the expected number of binary questions required to determine the value of X. It can be shown that the minimum number of binary questions required to determine X lies between $S_H(X)$ and $S_H(X+1)$.

Let's now introduce the definitions for joint and conditional entropy and mutual information.. These are key facets of the technology transfer models proposed.

2. Joint Entropy

Joint entropy $S(X,Y)$ of a pair of discrete random variables *(X,Y)* with a joint distribution *(X,Y)* can be considered to be a single vector-valued random variable. The *joint probability $p(X,Y)$* be defined as *$p(x,y)$* is the probability of a joint occurrence of event $X=x$ and event $Y=y$. This leads to

$$S_H(X,Y) = -\sum_{x \in \Xi} \sum_{y \in \Psi} p(x,y) \log(p(x,y) \qquad \text{(A.13)}$$

which can also be expressed as

$$S_H(X,Y) = -E \log p(X,Y) \qquad \text{(A.14)}$$

3. Conditional Entropy

The *conditional entropy* of a random variable given another is defined as the *expected value* of the entropies of the conditional distributions, averaged over the conditioning random variable. If *$(X,Y) \sim p(x,y)$*, the *conditional probability* is *$p(X/Y)$* of

outcome $X=x$ given outcome $Y=y$ for random variables (not necessarily independent). The conditional entropy $S(Y/X)$ is

$$S_H(Y|X) = \sum_{x \in \Xi} p(x) S_H(Y|X=x) \tag{A.15}$$

$$= -\sum_{x \in \Xi} p(x) \sum_{y \in \Psi} p(y|x) \log p(y|x) \tag{A.16}$$

$$= -\sum_{x \in \Xi} \sum_{y \in \Psi} p(y,x) \log p(y|x) \tag{A.17}$$

$$= -E_{p(x,y)} \log p(Y|X) \tag{A.18}$$

This is shown in the Venn diagram in Figure A-3. The mutual information is given as $I(X;Y)$.
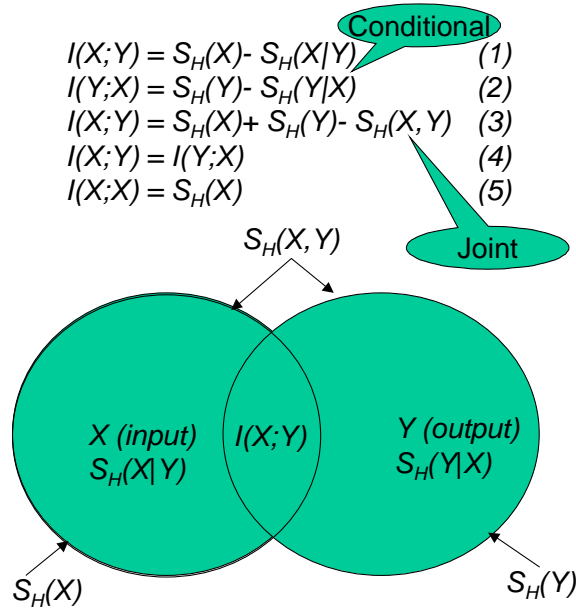
Figure A-3 Mutual Information, Joint and Conditional Entropy

Referring to Figure A-3for the models proposed, the entropy of the vocabulary of terms at time step k is the input entropy $S_H(X)$. The joint entropy $S_H(X,Y)$ is the cumulative entropy at time step $k+1$. The $S_H(Y)$ is the incremental contribution of the time step $k+1$. The mutual information, $I(X;Y)$, can be calculated from equation (3) in Figure A-3, given the data for the input entropy, the incremental contribution, and the joint entropy. Using Figure A-3, equations (2) and (3), the conditional entropy is readily computed.

Let's look at an example with a vocabulary of 4 terms {A,B,C,G} in a *di*-gram. We begin by building a matrix with the headings on the rows and columns being elements of the vocabulary. The frequency of the terms occurring together is given in the cell. When we have the term the AB, with the A in the row (this is the input X) then we have A appearing in the same message as B given in the column heading. In the models proposed, we are not concerned about the order of the terms, i.e. which precedes which, we are satisfied to know that a term appears with another. This is because we are using a message as represented by the records index terms. In free text implementations, without controlled indexing such as using the Internet, or data mining the case where A is the column, and B is the row heading, B precedes A. In our models we actually count the pairs, triples, etc and build the vocabulary since the languages of the technologies are generally small. Typically we see about 2000-3000 single terms.

Actually, there are sets of subsets {}, {A}, {B}, {C}, {G}, {AB}, {AC}, … {ABC}, {ACG}, … {ABCG} as possible "terms". To get the count of all of the permutations for triples, and quadruples, etc, the process can be repeated with the row headings, including pairs, and the columns singles. Similarly for quadruples, once the triples are computed, we can use the triples as the row headings and singles again as column headings. For our purposes, we have simplified the matrix for purposes of example. A {} preceding the column term could be arranged to imply that a new term has been added to the vocabulary in this time step.

Figure A-4 Example 1, Vocabulary Distribution

In Figure A-4, the entropy for the example 1 (*ex1*) marginal distribution of *X* is given as $S_H(X)_{ex1}$ (.25, .25, .25, .25) is 2 bits. The marginal distribution of *Y* is given as $S_H(Y)_{ex1}$ (.5, .25, .125, .125) is 1.75 or 7/4 bits. The conditional entropy of *Y* outcome given the *X*, is given as $S_H(X/Y)_{ex1}$ and is 1.625 or 13/8 bits. The conditional entropy of *X* outcome given *Y* given as $S_H(Y/X)_{ex1}$ is 1.375 or 11/8 bits. The joint entropy is from the probability of a joint occurrence. It is given as $S_H(X,Y)_{ex1}$ is 3.375 or 27/8 bits.

$$S_H(X,Y) \leq S_H(X) + S_H(Y) \qquad\qquad (A.19)$$

There is equality only in the case where *X* and *Y* are independent. In all of these equations, the entropy quantity on the left side increases as we choose probabilities on the right hand side more equally (recall Figure A-3). The mutual information *I(X;Y)* is computed

$$I(X;Y) = S_H(X) - S_H(X|Y) \qquad \text{(A.20)}$$

In this example, the mutual information is 2-1.625 = .375 or bits.

B. OPERATORS, EIGENVALUE SIGNIFICANCE, MARKOV CHAINS, ERGODIC PROCESSES

1. Operators and Eigenvalue Significance

In Chapter 3, we introduced the function $X_{k+1} = F(X_k)$ Let's put this more specifically in the terms of a distribution function $\rho(x)$ and provide an overview of the recurrence relation $\rho_{k+1}(x) = U\rho_k(x)$ (A.21)

The distribution function $\rho_{k+1}(x)$ after $(k+1)$ maps is obtained by the action of the operator U on $\rho_k(x)$, which is the distribution function after k maps. Let's consider what we all know from mathematics of periodic functions such as $\sin\left(\dfrac{2\pi x}{\lambda}\right)$. This function remains invariant when we add to the coordinate x the wavelength λ, as

$$\sin\frac{2\pi x}{\lambda} = \sin\frac{2\pi(x+\lambda)}{\lambda}\,.$$

Other periodic functions are $\cos\dfrac{2\pi x}{\lambda}$, or the more complex combination

$$e^{\frac{2\pi x}{\lambda}} = \cos\frac{2\pi x}{\lambda} + i\sin\frac{2\pi x}{\lambda}$$

With that notion in hand, what follows is a discussion by Prigogine for a quick review (Prigogine 1997, p92).

> An operator is a prescription on how to act on a given function; it may involve multiplication, division, differentiation, or any other mathematical operation. In order to define an operator, we define a function space. That is, we specify the domain, the types of functions it acts on, indicate whether they are continuous or bounded, and other characteristics as required. In general an operator U acting on a function $f(x)$ transforms it into a different function. For instance, if U is a derivative operator, $\dfrac{d}{dx}$, then $Ux^2=2x$. However, there are special functions known as *eigenfunctions* of the operator, which remain invariant when we apply U; they are multiplied only by a number, the *eigenvalue*. In the above example e^{kx} is an eigenfunction to which the eigenvalue k corresponds. A

fundamental theorem in operator analysis is states that we can express an operator in terms of its eigenfunctions and eigenvalues, both of which depend on its function space.

Physicists use Hilbert space in quantum mechanics. Prigogine goes beyond Hilbert space for operations in unstable dynamical systems.

Consider the "equations of motion"

$$x_{k+1} = x_k + \frac{1}{2},$$ modulo 1 (i.e. the numbers between 0 and 1)   See Figure A-5 for this simple periodic map. After two shifts we are back to the initial point.

i.e., $x_0 = \frac{1}{4}, x_1 = \frac{3}{4}, x_2 = \frac{3}{4} + \frac{2}{4} = \frac{5}{4} = \frac{1}{4}$

Instead of using individual points located by trajectories, we are using ensembles represented by the probability distribution $\rho(x)$. A trajectory corresponds to a set of ensembles where the coordinate x takes on a well-defined value for x_k, and the distribution function ρ is reduced to a single point. This can be written as

$$\rho_k(x) = \delta(x - x_k)$$

Here delta, δ, is a symbol for a function[1] that vanishes for all values of x except $x=x_k$. By using the distribution function ρ, the mapping can be expressed as a relation between $\rho_{k+1}(x)$ and $\rho_k(x)$. We can then write $\rho_{k+1}(x) = U\rho_k(x)$. Formally, $\rho_{k+1}(x)$ is known as the Perron-Frobenius operator acting on $\rho_k(x)$. The ensemble description must allow the trajectory description as a special case. So, we therefore have $\delta(x - x_{k+1}) = U\delta(x - x_k)$. This is just rewriting the equation of motion, as x_k becomes x_{k+1} after one shift.

---

[1] This is called the Dirac delta function. (Prigogine 1997 p33)

Simple geometrical construction that moves from initial point P_0 to the next point P_1 according to the map

$$x_{k+1} \to x_k + 1/2$$

We go from P_0 to P', then to P'' on the bisector, and from there to P_1. If we start with P_1, we come back to P_0
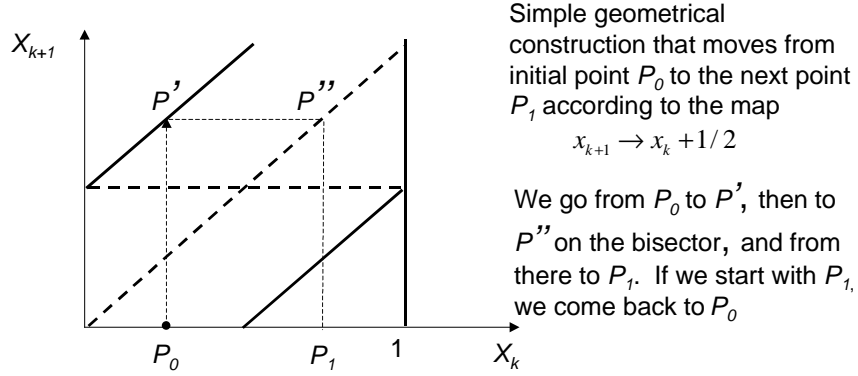
Figure A-5 Periodic Map (Source: Prigogine 1997, p82)

The simplest example of deterministic chaos is a Bernoulli map. In a Bernoulli map the value of a number doubles every time step, with the value of the number between 0 and 1. Consider the equation

$x_{k+1} = 2x_k$, modulo 1 (i.e. dealing with numbers between 0 and 1).

The equation of motion is again deterministic, since once we know x_k, the number x_{k+1} is determined. As the coordinate x is multiplied by two each time step, the distance between the tow trajectories, will be

$(2^k) = e^{k \log 2}$ modulo 1.

In terms of continuous time, t, this can be written as

$e^{t\lambda}$ with $\lambda = \log 2$

where λ is called the Lyapunov exponent. This shows the trajectories diverge exponentially, and is the signature of deterministic chaos. This is a dynamical process leading to randomness. What Prigogine does which is new, and we exploit here is the statistical formulation of the Bernoulli map, which links randomness to operator theory.

The explicit form of the evolution operator U obtaining

$$\rho_{k+1}(x) = U\rho_k(x) = \frac{1}{2}\left[\rho_k\left(\frac{x}{2}\right) + \rho_k\left(\frac{x+1}{2}\right) \right] \qquad (A.22)$$

This equation means that after $(k+1)$ iterations, the probability of $\rho_k(x)$ is determined by the values at points $\dfrac{x}{2}$ and $\dfrac{1+x}{2}$. As a consequence of the form of U, if ρ_k is a constant equal to α, ρ_{k+1} is equal to α, since $U\alpha=\alpha$. The uniform distribution $\rho=\alpha$, which corresponds to equilibrium, is the distribution function reached through the iteration of the shift, for $k\rightarrow\infty$.

On the other hand, we have the case when $\rho_k(x)=x$, here we have $\rho_{k+1}(x)=\dfrac{1}{4}+\dfrac{x}{2}$. In other words, $Ux=\dfrac{1}{4}+\dfrac{x}{2}$, where the U operator transforms the function x into a different function, $\dfrac{1}{4}+\dfrac{x}{2}$. We can find the eigenfunctions as defined above in which the operator reproduces the same function multiplied by a constant. In the example

$$U\left(x-\frac{1}{2}\right)=\frac{1}{2}\left(x-\frac{1}{2}\right) \tag{A.23}$$

the eigenfunction is therefore $x-\dfrac{1}{2}$ and the eigenvalue is $\dfrac{1}{2}$. If we repeat the Bernoulli map k times, we obtain

$$U^k\left(x-\frac{1}{2}\right)=\left(\frac{1}{2}\right)^k\left(x-\frac{1}{2}\right) \tag{A.24}$$

which moves toward 0 as $k\rightarrow\infty$. The contribution $\left(x-\dfrac{1}{2}\right)$ to $\rho(x)$ is therefore rapidly damped at a rate related to the Lyapunov exponent. This turns out to be a class of polynomials called Bernoulli polynomials. Denoted as which are eigenfunctions of U with eigenvalues of $\left(\dfrac{1}{2}\right)^k$, where k is the degree of the polynomial.

Prigogine is careful to emphasize the distinction between "nice" functions, and "singular" functions. These are also called generalized functions or distributions, which are not to be confused with probability distributions. The simplest singular function is the delta function $\delta(x)$. $\delta(x-x_0)$ is zero for all values where $x\neq x_0$, and infinite where $x=x_0$. Singular functions have to be used with nice functions. For example, if $f(x)$ is a continuous

function $\int dx f(x)\delta(x-x_0) = f(x_0)$ has a well defined meaning. In contrast the integral containing the product of singular functions, such as $\int dx \delta(x-x_0)\delta(x-x_0) = \delta(0) = \infty$, diverges and is meaningless.

Defining the operator U in terms of its eigenfunctions and eigenvalues is called the spectral representation of the operator U. There are the set of functions $B_k(x)$, the Bernoulli polynomials which are nice functions, but there is a second set, $\tilde{B}_k(x)$, which are formed by singular functions related to the derivatives of the δ-function. To obtain the spectral function of U and $U\rho$, we use both sets of eigenfunctions.

As a result, the statistical formulation of the Bernoulli map is applicable only to nice probability functions ρ and not to single trajectories that correspond to singular distribution functions represented by δ-functions. So the equivalence between the individual descriptions in terms of trajectories represented by δ-functions is broken. For the continuous distribution ρ, Prigogine obtained results that go beyond trajectory theory.

We can calculate the rate of approach to equilibrium and therefore to an explicit dynamical formulation of irreversible processes that take place in a Bernoulli map. Probability distribution takes into account the complex microstructure of the phase space.

When using both the $B_k(x)$, which are nice functions, and the second set, $\tilde{B}_k(x)$ which are singular functions, Prigogine moves from simple Hilbert space to a rigged Hilbert space, or Gelfand space. He obtains an irreversible spectral representation of the Perron-Frobenius operator as it applies exclusively to nice probability distributions, and not individual trajectories.

2.   Bakers Transformation

The attractor for the Bernoulli shift with an irrational initial condition x_0 is the unit interval, with fractal dimension 1 (see Farmer 1983, Baker 1990 for a discussion of fractal dimensions). The attractor for the bakers' transformation is the unit square, with fractal dimension 2. The dissipative bakers transformation is given (with a>0) by combining the Bernoulli shift

$$x_{k+1} = 2x_k, \quad \text{modulo 1} \qquad\qquad (A.25)$$

with the mapping

$$y_{k+1} = \begin{cases} ay_k & 0 \le x_k < \dfrac{1}{2} \\ ay_k + \dfrac{1}{2} & \dfrac{1}{2} \le x_k \le 1 \end{cases} \qquad (A.26)$$

The transformation is dissipative for $a<1/2$, because

$$J = \frac{\partial(x_{k+1}, y_{k+1})}{\partial(x_k, y_k)} = \begin{vmatrix} 2 & 0 \\ 0 & a \end{vmatrix} = 2a \qquad (A.27)$$

See McCauley (McCauley 1993, p132) for further discussion.

The Bernoulli map is not an invertible system. Because the arrow of time exists, we have to describe the emergence of irreversibility in invertible dynamical systems. The bakers map or bakers transformation is a generalization of the Bernoulli map (Prigogine 1997), (Tabor 1989), (Baker 1990), (Farmer 1983). Take a square that has sides of length 1. First flatten the square into a rectangle whose length is 2; then cut it in half and build a new square. This is illustrated in Figure A-6 shows an area preserving transformation similar to a baker rolling out dough. Since the distance between two points along the horizontal coordinate doubles with each transformation, it will be multiplied by 2^k after k transformations. Rewriting 2^k as $e^{k \log 2}$, as the number k of transformations of measure time, the Lyapunov exponent is exactly as in the Bernoulli map. There is also a second Lyapunov exponent with a negative value $-\log 2$, which corresponds to the contracting direction of y.

Prigogine and others show when relating the bakers transformation in the representation of a Bernoulli shift, the information contained in the initial condition contains the entire past history and future. Again from Prigogine,

> The critical point is that for typical, irrational initial coordinates x_0, y_0 associated binary representations can yield a doubly infinite sequence ($k=-\infty$, and $k=+\infty$) as random as a fair coin toss. Thus a completely deterministic dynamical system can yield results that appear completely random. The bakers transformation also has the property of all dynamical systems, recurrence. The bakers transformation is *invertible, time reversible, deterministic, recurrent* and *chaotic.*

Bakers Transformation

$$\begin{bmatrix} x_{k+1} \\ y_{k+1} \end{bmatrix} = \begin{bmatrix} 2x_k \\ y_k/2 \end{bmatrix}, \qquad 0 \le x_k < \frac{1}{2}$$

- Repeated doublings in the *x* direction and halving in the *y* direction leads to rapid mixing.

$$= \begin{bmatrix} 2x_k - 1 \\ y_k/2 + \frac{1}{2} \end{bmatrix}, \quad \frac{1}{2} \le x_k \le 1$$

- The mapping is completely reversible. Run backwards, the doubling occurs in the *y* direction and halving occurs in the *x* direction
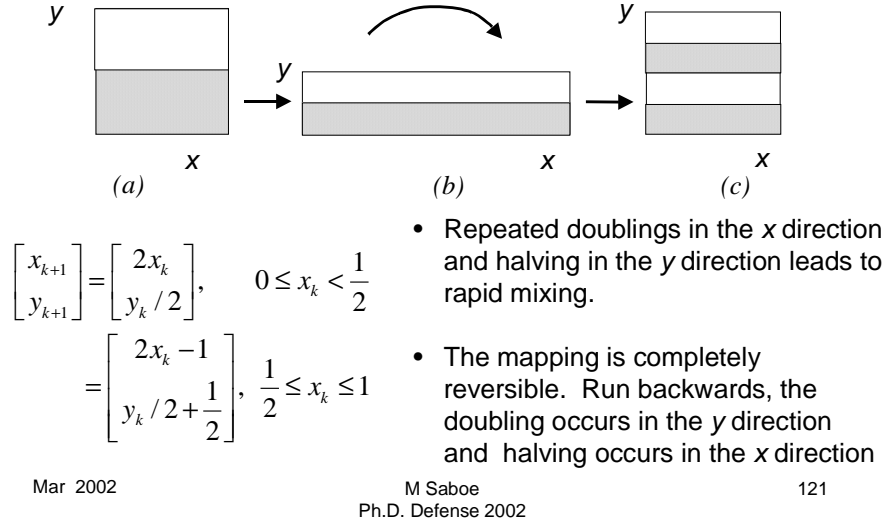
Figure A-6 Bakers Transformation

For the baker map there is a new element compared to the Bernoulli map (Prigogine 1997, p104). Prigogine shows that the Perron-Frobenius equation can be applied to both the future and the past.

$$\rho_{k+1} = U \rho_k$$

and

$$\rho_{k-1} = U^{-1} \rho_k$$

Here U^{-1} is the inverse of *U*. For irreducible spectral representations, there is an essential difference between past and future.

Prigogine's research has also shown that irreversibility is linked only to Lyapunov time for general irreversible phenomena such as diffusion and various other transport processes.

C. MARKOV CHAINS, ERGODIC PROCESSES

These sections provide a stand alone reference following Bronson (Bronson 1982).

1. Markov Process

A Markov process is a process where the future evolution of a state depends only on the present state. A Markov process (Bronson 1982, p224) consists of a set of objects and a set of states such that

- at any given time an object must be in a stare (distinct objects need not be in distinct states;

- the probability that an object moves from one state to another state (which may be the same as the first state) in one time period depends only on those two states.

- The integral number of time periods, past the moment when the process is started represent the stages of the process, which may be finite or infinite. If the number of states is finite or countably infinite, the process is called a Markov chain. A finite Markov chain has a finite number of states.

- P_{ij} denotes the probability of moving from state i to state j in one time step. For an N state Markov chain (where N is a fixed positive integer), the N x N matrix $\mathbf{P}=[p_{ij}]$ is the stochastic or transition matrix associated with the process. The elements in each row of $\mathbf{P}$ must sum to one (unity).

Theorem 19.1 (Bronson 1982 p224) states: Every stochastic matrix has 1 as an eigenvalue (possibly multiple), and none of the eigenvalues exceeds 1 in absolute value. Because of the way that $\mathbf{P}$ is defined, it is convenient to indicate *N*-dimensional vectors as row vectors, with matrices operating on them from the right. According to theorem 19.1 above, there exists a

vector $\rho \neq 0$ such that

$$\rho \mathbf{P} = \rho$$

This left eigenvector is called a fixed point of **P.**

Powers of stochastic matrices are denoted by n. The nth power of matrix **P** is indicated by $\mathbf{P}^n \equiv [p_{ij}^{(n)}]$. If **P** is stochastic,, then $p_{ij}^{(n)}$ represents the probability that an object moves from state i to state j in n time steps. It follows that $\mathbf{P}^n$ is also a stochastic matrix. We denote the proportion of objects in the state i at the end of the nth time step by $\rho_i^{(n)}$, and designate

$$\rho^{(n)} \equiv [\rho_1^{(n)}, \rho_2^{(n)}, ..., \rho_N^{(n)}]$$

is the distribution vector for the end of the nth time step. Similarly,

$$\rho^{(0)} \equiv [\rho_1^{(0)}, \rho_2^{(0)}, ..., \rho_N^{(0)}]$$

represents the proportion of the objects in each state at the beginning of the process. $\rho^{(n)}$ is related to $\rho^{(0)}$ by the equation

$$\rho^{(n)} = \rho^{(0)} \mathbf{P}^n \qquad\qquad \textit{(Bronson 19.1)}$$

In writing theorem *19.1* the proportion of the objects in state i that make the transition to state j are implicitly identified with the probability p_{ij}.

2.   Ergodic Process

Again following Bronson (Bronson 1982 p 225) we define the properties required for an ergodic process in terms of egrodic and regular matrices..

A stochastic matrix **P** is ergodic if $\lim_{n \to \infty} \mathbf{P}^n$ exists; that is, if each $p_{ij}^{(n)}$ has a limit as $n \to \infty$. The limit matrix is denoted by necessity by **L**. The components of $\rho^{(\infty)}$, defined by the equation

$$\rho^{(\infty)} = \rho^{(0)} \mathbf{L} \qquad\qquad \textit{(Bronson 19.2)}$$

are the limiting state distributions and represent the approximate proportions of objects in the various states of the Markov chain after a large number of time steps.

Theorem 19.2 (after Bronson 1982, p225) states

A stochastic matrix is ergodic if and only if the eigenvalue λ of magnitude 1 is 1 itself and, $\lambda=1$ has multiplicity k, there exists a k linearly independent (left) eigenvectors associated with this eigenvalue.

Theorem 19.3 (after Bronson 1982, p225) states

If *every* eigenvalue of a matrix **P** yields linearly independent (left) eigenvectors in number equal to its multiplicity, then there exists a nonsingular matrix **M**, whose left eigenvectors of **P**, such that **D≡MPM^{-1}** is a diagonal matrix. The diagonal elements of **D** are the eigenvalues of **P**, repeated according to multiplicity. The convention is adopted of positioning the eigenvectors corresponding to $\lambda=1$ above all other eigenvectors in **M**. Then the diagonalizable, ergodic, N x N matrix **P** with $\lambda=1$ of multiplicity k, the limit matrix **L** may be calculated as

$$\mathbf{L} = \mathbf{M}^{-1}(\lim_{n\to\infty} \mathbf{D}^n)\mathbf{M} = \mathbf{M}^{-1}\begin{bmatrix} 1 & & & & & & & & \\ & 1 & & & & & & & \\ & & \cdot & & & & & & \\ & & & \cdot & & & & & \\ & & & & \cdot & & & & \\ & & & & & 1 & & & \\ & & & & & & 0 & & \\ & & & & & & & \cdot & \\ & & & & & & & & \cdot \\ & & & & & & & & & \cdot \\ & & & & & & & & & & 0 \end{bmatrix}\mathbf{M}$$

(Bronson 19.3)

The diagonal matrix on the right has k 1's and $(N-k)$ 0's on the main diagonal. A stochastic matrix is regular if one of its powers contains only positive elements.

Theorem 19.4 (From Bronson 1983, p225) states

If a stochastic matrix is regular, then 1 is an eigenvalue of multiplicity one, and all other eigenvalues λ_i satisfy $|\lambda_l|<1$.

Theorem 19.5 (From Bronson 19983, p225) states

A regular matrix is ergodic.

If **P** is regular, with limit matrix **L**, then the rows of **L** are identical with one another, each being the unique left eigenvector of P associated with the eigenvalue $\lambda=1$ and having the sum of its components equal to unity. Denote this eigenvector by $\mathbf{E}_1$. It follows directly from *(Bronson 19.2)* that **P** is regular, then regardless of the initial distribution of

$$\rho^{(\infty)}= \mathbf{E}_1 \qquad\qquad\qquad \textit{(Bronson 19.4)}$$

Figure A-7 and Figure A-8 provide an example of the state transition rules in a communication context after Shannon, and an example of a two state Markov chain.

Markov Processes
State Transition Rules

- Stochastic processes known as Markov process
- There exists a finite number of possible states in the system *S1, S2, , Sn*
- There is a set of transition probabilities;
    - *Pi (j)* the probability that if the system is in state *Si* it will go next to state *Sj*
- State will correspond to a "residue" of influence from preceding messages
- The pro   ergodic -- i.e. roughly this means every state
    properties   neous   Shannon 1949

                            or any node
                       e inputs and output properties
                       in this case -- will equal 1

Figure A-7 Markov Process State Transition Rules

Example Two State Markov Chain with Probability Transition Matrix

$$P = \begin{bmatrix} 1-\alpha & \alpha \\ \beta & 1-\beta \end{bmatrix}$$
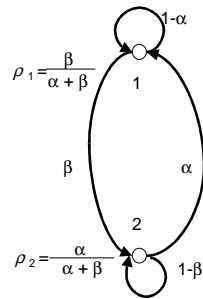
- Stationary distribution
    - represented by vector β
    - components are stationary probabilities of
    - state 1 and state 2 respectively

- Stationary Probability found by solving by $\rho P = \rho$
    - or balancing the probabilities

$\rho_1 = \dfrac{\beta}{\alpha + \beta}$

$\rho_2 = \dfrac{\alpha}{\alpha + \beta}$

- For Stationary distribution the net probability flow across any cut - set in the state transition graph is 0

$$\rho_1 \alpha = \rho_2 \beta \qquad \text{Since } \rho_1 + \rho_2 = 1$$

$$\rho_1 = \frac{\beta}{\alpha + \beta} \qquad \rho_2 = \frac{\alpha}{\alpha + \beta}$$

- Entropy at state X_n at time n is

$$S(X_n) = \begin{bmatrix} \dfrac{\beta}{\alpha + \beta} & , & \dfrac{\alpha}{\alpha + \beta} \end{bmatrix}$$

Figure A-8 Example of Two State Markov Chain

D.   SYMBOLIC DYNAMICS AND INFORMATION

We consider a system in discrete state space. The development, which follows, is structured closely to the exposition by Prigogine (Prigogine 1987, p183) on symbolic dynamics and information.

Establish a probability distribution underlying a process.   Set up balance equations that counts the processes leading the system to state Q and the processes removing it from the state (Prigogine 1987 p153).

We get

$$\frac{d\ prob(Q,t)}{dt} = \text{(contribution of transitions to state } Q \text{ per unit time)}$$

$$- \text{(contribution of transitions from state } Q \text{ per unit time)}$$

$$\equiv R_+(Q) - R_-(Q)$$

(Prigogine 1987, eqn4.9)

which becomes a problem of determining the transition rates R_+ and R_-. The system must satisfy conditions of a detailed balance following the constraint conditions, similar to that of thermodynamics, or similarly Markov processes above. So if we decompose R_+ and R_- into the elementary processes taking place in the system,

$$R_\pm = \sum_k r_{k,\pm}$$

The following local condition must be satisfied.

$$(r_{k,+})_{equil} = (r_{k,-})_{equil} \qquad \textit{(Prigogine 1987, eq 4.10)}$$

These relations must in turn be compatible with the form of the probability distribution in the state of equilibrium, which is known from statistical mechanics. The limiting case of such a distribution is a Poisson distribution. Einstein showed at equilibrium, that the probability of fluctuations is entirely determined by thermodynamic quantities. In an isolated system (i.e. in a control volume, the inversion of Boltzmann's formula yields

$$S = k_b \ln(\text{ number of molecular arrangements compatible with a given energy value})$$

Which we know from Shannon's *theorem 2*, that Boltzmann's constant k_b, and the natural log can be eliminated and converted to log_2 respectively (Shannon 1948, p11, p1) for the measure of entropy in information units. So since Jaynes (Jaynes 1956, 1956a) developed the relationship of information and communication theory to statistical mechanics, we can invert the equation and write

$$p_{equil} \sim e^{\Delta S} \qquad \textit{(After Prigogine 1987 eq 4.11)}$$

where ΔS is the change in entropy due to fluctuation, $\Delta S = S(Q) - S(Q_{equil})$.

Prigogine also requires that the *(Prigogine 1987, eq4.9),*

> …in a limiting sense, must reduce to evolution dealt with in the deterministic description. We expect the macroscopic observations will yield values representative of the most probable state in a physical system. Looking at this mathematically, we would expect the same for a communication channel, i.e. that the peaks of *P(Q,t)* be solutions to deterministic equations.

If the system is uni-modal Figure A-9, which is our case in an evolutionary system, this implies that the equation for the mean value is close to the deterministic equation, the correction is essentially proportional to the inverse power of the size of the system.
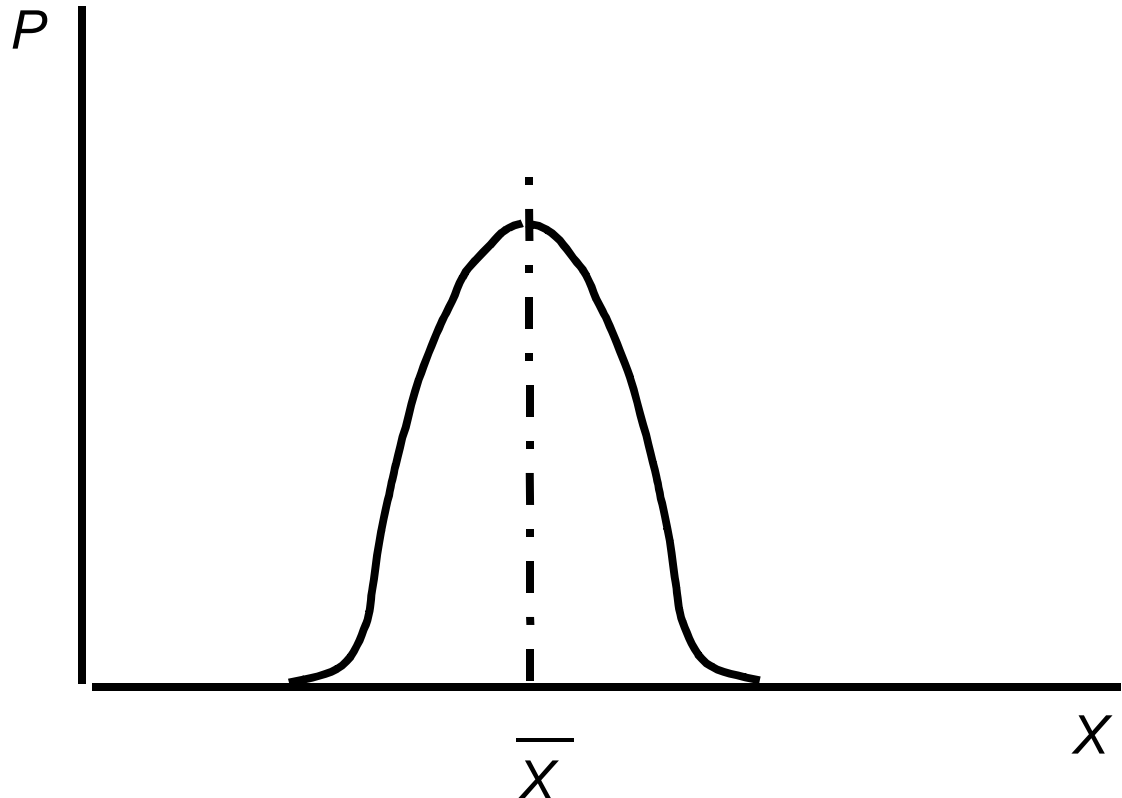
Figure A-9 Uni-modal Distribution

Let $\{Q_I\}$ ($I = 1,2, ..,$) be accessible states of a system. These states of $\{Q_I\}$ are chosen so that the time evolution defines a Markov process.

APPENDIX B EQUATIONS AND SAMPLE CALCULATIONS

This section provides the equations used in completing the calculations required. The tables and data were extracted from a supporting document (Behnke 2001) which describes the calculations. This represents example data, not necessarily the functions used for the data sets in the final dissertation. For example, the power function is explained as opposed to the linear relationship of entropy verse time step.

1.   Entropy Calculation Equations and Example:

The formula used in this calculation is the following:

$$-(\text{probability of term usage}) * \log_2 (\text{probability of term usage}) \qquad (B.1)$$

Probability of term usage is the cumulative number of a single term's instances up to the given time interval divided by the number of terms instances for all the terms up to the given time interval. The following two tables give an example of the calculation:

This section provides the equations used in completing the calculations required.

Entropy Calculation Equations and Example:

The formula used in this calculation is the following:

-(probability of term usage) * $\log_2$ (probability of term usage)   (B.2)

Probability of term usage is the cumulative number of a single term's instances up to the given time interval divided by the number of terms instances for all the terms up to the given time interval. The following two tables give an example of the calculation:

Term	1989	1990	1991
A – # of instances	2	3	5
B – # of instances	1	5	18
Local sum	3	8	23
Cumulative sum	3	11	34

Table B-1 Sample Calculation Data

Term			
Entropy of A	$-(2/3) * \log_2 (2/3)$ = 0.3900	$-(5/11) * \log_2 (5/11)$ = 0.5170	$-(10/34) * \log_2 (10/34)$ = 0.5193
Entropy of B	$-(1/3) * \log_2 (1/3)$ = 0.5283	$-(6/11) * \log_2 (6/11)$ = 0.4770	$-(24/34) * \log_2 (24/34)$ = 0.3547
Cumulative entropy	0.9183 (a + b)	0.9940	0.8740

Table B-2 Sample Calculation Equation with Data

2. Predicted Entropy Calculation

The predicted entropy value for a time interval is calculated using the trend-line power equation (the least squares fit through points):

$$y = cx^b \qquad\qquad\qquad (B.3)$$

where *c* and *b* are constants. The time interval replaces *x*. An example from the Ada dataset follows in Table B3.

Percent Error of Actual vs. Predicted

Percent error of actual vs. predicted is calculated using the formula below and an example follows in Table B3.

$$\text{Error} = \frac{\text{Predicted} - \text{Actual}}{\text{Actual}} \qquad \text{(B.4)}$$

Time T	Slice	Actual: Cum Entropy	Error (Act vs. Pred) $y = 4.7404x^{.1489}$	Predicted: 5 years of data
1	1979	4.48385619	5.71%	4.48385619
2	1980	5.406900167	-2.86%	5.406900167
3	1981	5.805013635	-3.93%	5.805013635
4	1982	6.057909749	-3.94%	6.057909749
5	1983	6.082181413	-1.11%	6.082181413
6	1984	6.106601538	1.19%	6.179377493
7	1985	6.355700897	-0.53%	6.321976128
8	1986	6.52682382	-1.21%	6.44815784
9	1987	6.549095131	0.19%	6.561546835
10	1988	6.611519798	0.80%	6.664665264

Table B-3 Predicted Entropy Calculation and Error Example

Note: First 5 intervals under the predicted column are copied from actual.

3. Time Interval Derivative Calculation

The Du(T) and Du_(T-c) calculations are based on the derivative of the trend-line's equation from the cumulative entropy graph. The derivative of the trend-line's equation is taken and then the time interval replaces x. The following is the equation used:

$$\frac{dy}{dx}\left[y = cx^b\right] = cbx^{(b-1)} \qquad \text{(1.4)}$$

A usage example from the Ada dataset follows in Table B4.

Table B-4

	Time T			
1	0.70152			
2	0.388653426	0.70152		
3	0.275126706	0.388653426	0.70152	
4	0.215320284	0.275126706	0.388653426	
5	0.178040011	0.215320284	0.275126706	
6	0.152424645	0.178040011	0.215320284	0.70152
7	0.133664638	0.152424645	0.178040011	0.388653426
8	0.11929092	0.133664638	0.152424645	0.275126706
9	0.107900992	0.11929092	0.133664638	0.215320284
10	0.098637046	0.107900992	0.11929092	0.178040011
11	0.090943882	0.098637046	0.107900992	0.152424645
12	0.084445719	0.090943882	0.098637046	0.133664638
13	0.078878805	0.084445719	0.090943882	0.11929092
14	0.074052371	0.078878805	0.084445719	0.107900992
15	0.069824897	0.074052371	0.078878805	0.098637046

Table B-4 Time Interval Derivative Calculation Example

<u>Note.</u> $y = 4.7404x^{0.1489}$

$du(T) = (4.740*0.148)*T^{\wedge}(0.148-1)$

$du(T-c) = (4.740*0.148)*(T-c)^{\wedge}(0.148-1)$

4.   Lambda Calculation

The lambda calculation is dependent on the time interval derivative calculations.

The equation to calculate lambda is:

$$\lambda_0 = (\beta f'(x))^{1/3} \tag{B.5}$$

Where $f'(x)$ is substituted with:

$$f'(x_0) = \frac{\partial y}{\partial x} = \frac{\dfrac{du_{(t-c)}}{dt}}{\beta\dfrac{du_{(t-c)}}{dt} + \dfrac{du_{(t)}}{dt}} \tag{B.6}$$

substituting for $f'(x)$ we get :

$$\lambda_0 = \left(\beta \frac{\dfrac{du_{(t-c)}}{dt}}{\beta \dfrac{du_{(t-c)}}{dt} + \dfrac{du_{(t)}}{dt}} \right)^{1/3} \tag{B.7}$$

The values from the time interval derivative equation (1.4) are placed into (1.7) with varying β values (e.g. 0.1, 0.2, 0.5, 0.75). Table B5 shows an example of the lambda calculation.

Time T	Cum Entropy	Du_(T)	C_y_10%	Lambda_β 10%_y	β 10%_y
1	4.48385619	0.70152			
2	5.406900167	0.388653426	4.87216694	0.534733226	0.1
3	5.805013635	0.275126706	5.306648158	0.498365477	0.1
4	6.057909749	0.215320284	5.574025253	0.483884497	0.1
5	6.082181413	0.178040011	5.60612135	0.476060063	0.1
6	6.106601538	0.152424645	5.635448868	0.47115267	0.1
7	6.355700897	0.133664638	5.887915573	0.467785324	0.1
8	6.52682382	0.11929092	6.061493127	0.465330693	0.1
9	6.549095131	0.107900992	6.085633441	0.46346169	0.1
10	6.611519798	0.098637046	6.14952886	0.461990938	0.1
11	6.64290191	0.090943882	6.182098576	0.460803334	0.1
12	6.725985485	0.084445719	6.266161229	0.459824255	0.1

Table B-5 Lambda Calculation Example

Note. "C_y_10%" is found from "Cum Entropy" minus "Lambda_ β 10%_y"

5.   Lyaponuv Exponent Calculation

The Lyaponuv exponent calculation depends on the trend-line equation from the map of entropy at time steps k and k+1. The derivative is taken the same way as in equation (1.4). Once the derivative is found the time interval is replaced for x. A usage example from the Ada dataset follows in Table B6.

Time T	Cum Entropy K	Cum_K+1	Lyaponuv Exp J'(k,k+1) = 0.724*1.720 k^(0.724-1)
1	4.48385619	5.406900167	0.823021444
2	5.406900167	5.805013635	0.781579695
3	5.805013635	6.057909749	0.766403215
4	6.057909749	6.082181413	0.757435961
5	6.082181413	6.106601538	0.756600505
6	6.106601538	6.355700897	0.755764222
7	6.355700897	6.52682382	0.74747023
8	6.52682382	6.549095131	0.742009203
9	6.549095131	6.611519798	0.741311905
10	6.611519798	6.64290191	0.739373453
11	6.64290191	6.725985485	0.738407755
12	6.725985485	6.817516503	0.735878946

Table B-6   Lyapunov Exponent Calculation Example

<u>Note.</u> $y = 1.7208x^{0.7241}$

$dx = 0.724*1.720)K^{(0.724-1)}$

APPENDIX C SAMPLE DATA

Sample data is on the CD under the directory labeled

\Entropy data analysis

THIS PAGE INTENTIONALLY LEFT BLANK

APPENDIX D TECH OASIS INTERFACE SOURCE CODE

This section contains the source code of the Tech OASIS interface. This source was written by Matt Behnke as partial contribution to his Masters Degree in Software Engineering in support of Dr. Michael S. Saboe. They can be reached at saboem@tacom.army.mil and behkneM@tacom.army.mil.

The source code is on the CD under the \Entropy data analysis\ directory

```
'------------------------------------------------------------------------
'  Script: cumEntropy.tmf
'  Author: Matt Behnke
'  Created: 9/10/01
'  Description: Tech OASIS script that prompts the user to select the data
'          field and time field to use in calculating the cumulative entropy.
'          The script exports the co-occurrence matrix of the two fields into
'          Microsoft Excel and then calls an excel macro to finish the manipulation
'          of the raw data to create a summary and graphs.
'
'------------------------------------------------------------------------

Option Explicit

'declare variables
  dim nStatus, strDataField, strTimeField, arrayGroupNames
  dim exApp, strView, strDirectoryPath

'prompt for and get user input (R1.1, R1.2)
  msgbox("Select data field to compute entropy on  ")
  nStatus = Dataset.PromptForField(strDataField)
  msgbox("Select time field that contains intervals as groups ")
  nStatus = Dataset.PromptForField(strTimeField)

'check to make sure there are groups inside the user selected time field (R1.3)
  nStatus = Dataset.GetGroupNames(strTimeField, arrayGroupNames)
  If(IsArray(arrayGroupNames)) Then

  Else
          MsgBox("There are no Groups in the time field! Program ending...")
          Stop
  End if

'Open Excel Workbook (R1.5)
  Set exApp = CreateObject("Excel.Application")
  exApp.Visible = True
  exApp.Workbooks.Add
```

```
          call createMatrix(strDataField, strTimeField)
          call runExcelMacro


          '---------------------------------------------------------------------------
          ' Function: createMatrix
          ' Author: Matt Behnke
          ' Created: 9/10/01
          ' Description: 1) Creates the co-occurrence matrix of data field (rows) X time
          '              field (columns) (R1.4)
          '           2) Exports this matrix to the opened excel file. (R1.6)
          ' Inputs: strDataField - user selected data field
          '          strTimeField - user selected time field
          ' Outputs: none
          '---------------------------------------------------------------------------
          Sub createMatrix(strDataField, strTimeField)

          'create and sort matrix
          nStatus                                                                      =
View.CreateMatrix(strDataField,"UNGROUPED",strTimeField,"GROUPED","COOCCURENCE",strVie
w)
          nStatus=Matrix.Sort("ROW",2,"DESCEND")

          nstatus=Matrix.SelectAll()
          nstatus=Matrix.CopySelection()

          'paste into excel
          exApp.ActiveSheet.Paste

          end sub


          '---------------------------------------------------------------------------
          ' Function: runExcelMacro
          ' Author: Matt Behnke
          ' Created: 9/10/01
          ' Description: Calls the excel macro "Cumulative" inside "cumEntropy.xls"
          '           located in the vantagepoint (Tech OASIS) macros directory.
          '           The macro finishes the calculation of cumulative entropy. (R1.7)
          ' Inputs: none
          ' Outputs: none
          '---------------------------------------------------------------------------
          Sub runExcelMacro()

          nStatus=App.GetPath(strDirectoryPath)
          strDirectoryPath=strDirectoryPath & "Macros\"
          exApp.WorkBooks.Open(strDirectoryPath & "cumEntropy")
          exApp.Windows(2).Activate
          exApp.Application.Run "cumEntropy.xls!Cumulative"
          exApp.visible=true
          exApp.WorkBooks(2).Close

          end sub
```

APPENDIX E DATA ANALYSIS SOURCE CODE

This source was written by Matt Behnke as partial contribution to his Masters Degree in Software Engineering in support of Dr. Michael S. Saboe. They can be reached at saboem@tacom.army.mil and behkneM@tacom.army.mil.

The source code is on the CD under the \Entropy data analysis\ directory

```
       This section contains the source code used to complete all of the
data analysis.  '----------------------------------------------------
-----
       ' MACRO: AffiliationMacro
       ' Author: Matt Behnke
       ' Created: 11/5/01
       '
       '
       '----------------------------------------------------------------------

       'GLOBAL VARIABLES

       Dim technologyName As String              'name of the dataset (ada, java, etc)
       Dim stepInterval As String            'the time between time steps (months, years)
       Dim currFilename As String             'the name of the spreadsheet file
       Dim dataSheet As String            'sheet that contains the matrix of affiliations
       Dim descriptorMatrixSheet As String           'sheet that contains the matrix of terms (X)
       Dim descriptorMatrixSheetY As String               'sheet that contains the matrix of terms
(opposite of X)
       Dim worldEntropySheet As String            'sheet that contains world entropy
       Dim worldEntropySheetY As String             'sheet that contains world entropy y (opposite
of X)
       Dim affiliationDescMatrix As String           'sheet that associates terms to affiliations

       'CONSTANTS
         Private Const HYP3_FIT As Integer = 0
         Private Const EXP3_FIT As Integer = 1
         Private Const POW2_FIT As Integer = 2


       '----------------------------------------------------------------------
       ' Sub: DistributeAffiliations
       ' Author: Matt Behnke
       ' Created: 11/5/01
       ' Description: The sub routine that calls all the sub routines for the affiliation distribution
```

309

```
' inputs:
'
' Outputs:
'----------------------------------------------------------------------
Sub DistributeAffiliations()

    currFilename = Application.ActiveWorkbook.Name
    dataSheet = ActiveSheet.Name

'sheet name constants
    descriptorMatrixSheet = "descriptor_data_X"
    descriptorMatrixSheetY = "descriptor_data_Y"
    worldEntropySheet = "World_Cumulative_Entropy_X"
    worldEntropySheetY = "World_Cumulative_Entropy_Y"
    affiliationDescMatrix = "descriptor_matrix_affil"

    technologyName = InputBox("Enter the name of the technology.")
    stepInterval = InputBox("Enter the time between time steps")

    Call formatSheetForPrint
    Call CopyMathCadObj

'put the cumulative values on the sheets:
    Call CalcCumulative(dataSheet)  'datasheet has the num records each affilation produced over

time

    Call CalcCumulative(descriptorMatrixSheet)
    Call CalcCumulative(affiliationDescMatrix)
    Call CalcCumulative("Affiliation_authors")

'determine the num of records in each band
    Call AffiliationDistribution
'use the summary sheet created by Affiliation_Distribution to graph the distributions of each band:
    Call CopyDistributionGraph

'compute world entropy (input, output) obsolete
    'Call ComputeEntropy(descriptorMatrixSheet, worldEntropySheet)

'create descriptor data y sheet from descriptor data x sheet:
    Call CreateDescriptorDataY("descriptor_data_X", "descriptor_data_Y")

'compute world entropy sheets x and y (input, output)
    Call ComputeEntropy("descriptor_data_X", "World_Cumulative_Entropy_X", 1)
    Call ComputeEntropy("descriptor_data_Y", "World_Cumulative_Entropy_Y", 2)

'fills the band stats of the world:

    Call FillBandStats("World")

'compute nu and psi for the world:
    Call v_calc_v_psi_sheet("World")
```

310

```
'----------------BANDS-----------------------------------

'fill the band with the affiliations and their number of publications that fit the number of
'publications range for that band determined by Affiliation_Distribution:
   Call FillBand("A_Band")
'fill band stats:
   Call FillBandStats("A_Band")
'create the matrix of affiliation with author instances
   Call FillBandAuthors("A_Band")
'calculate nu and psi:
   Call v_calc_v_psi_sheet("A_Band")
'determine the matrix of terms and the number of instances for the band
   Call FillBandTerms("A_Band")
'compute the entropy of the terms in the band
   Call FillBandTermsEntropy("A_Band")
'create a summaty of band.. num of publications, authors, terms, entropy:
   Call affiliationBandSummary("A_Band")


   Call FillBand("B_Band")
   Call FillBandStats("B_Band")
   Call FillBandAuthors("B_Band")
   Call v_calc_v_psi_sheet("B_Band")
   Call FillBandTerms("B_Band")
   Call FillBandTermsEntropy("B_Band")
   Call affiliationBandSummary("B_Band")


   Call FillBand("C_Band")
   Call FillBandStats("C_Band")
   Call FillBandAuthors("C_Band")
   Call v_calc_v_psi_sheet("C_Band")
   Call FillBandTerms("C_Band")
   Call FillBandTermsEntropy("C_Band")
   Call affiliationBandSummary("C_Band")


   Call FillBand("D_Band")
   Call FillBandStats("D_Band")
   Call FillBandAuthors("D_Band")
   Call v_calc_v_psi_sheet("D_Band")
   Call FillBandTerms("D_Band")
   Call FillBandTermsEntropy("D_Band")
   Call affiliationBandSummary("D_Band")


   Call entropySummary  'for the world
   Call affiliationSummary 'for the world
   Call affiliationSummaryPart2 'copies graphs and computes
```

```
        Call affiliationSummaryPart3 'temp and pressure...

        Call CopyABCDGraph 'copy the abcd learning curve graphs
        Call fillMonthsRowTrigger
        Call CopyBandSummaryGraphs("A_Band") 'entropy summary graphs
        Call CopyBandSummaryGraphs("B_Band")
        Call CopyBandSummaryGraphs("C_Band")
        Call CopyBandSummaryGraphs("D_Band")
        Call CopyBandSummaryGraphs("World")
End Sub


'-------------------------------------------------------------------------
'  Sub: AffliationDistribution
'  Author: Matt Behnke
'  Created: 11/5/01
'  Description: figures out the division of bands, and the number of affiliations per band
'  inputs:
'
'  Outputs:
'-------------------------------------------------------------------------
Sub AffiliationDistribution()
        Sheets.Add After:=Worksheets(Worksheets.Count)

        numRows = CountRows(dataSheet, 1)
        Sheets(Worksheets.Count).Select
        ActiveSheet.Name = "Distribution"

        Cells(1, 1) = "Statistics"

        Cells(2, 1).FormulaR1C1 = "Mean"
        Cells(2, 2).Formula = "=AVERAGE(" & dataSheet & "!A2:A" & numRows & ")"

        Cells(3, 1) = "Stdev"
        Cells(3, 2).Formula = "=STDEV(" & dataSheet & "!A2:A" & numRows & ")"

        Cells(4, 1) = "Sum"
        Cells(4, 2).Formula = "=SUM(" & dataSheet & "!A2:A" & numRows & ")"

        Cells(5, 1) = "Count"
        Cells(5, 2).Formula = numRows - 1

        Cells(2, 5).Formula = "Calculate Bands"
        Cells(3, 5).Formula = "Band_D"
        Cells(3, 6).Formula = "Band_C"
        Cells(3, 7).Formula = "Band_B"
        Cells(3, 8).Formula = "Band_A"
        Cells(4, 4).Formula = "from"
        Cells(5, 4).Formula = "to"

        Cells(4, 5).Formula = "0"                    'Band D from
```

```vba
        Cells(5, 5) = "=ROUND(B2+B3,3)"           'band d to

        Cells(4, 6) = "=ROUND(B2+B3,3)"           'band c from
        Cells(5, 6) = "=ROUND(B2+B3*2,3)"          'band c to

        Cells(4, 7) = "=ROUND(B2+B3*2,3)"          'band b from
        Cells(5, 7) = "=ROUND(B2+B3*3,3)"          'band b to

        Cells(4, 8) = "=ROUND(B2+B3*3,3)"          'band a from

    'bin labels
      Cells(7, 1) = "Bin"
      Cells(7, 2) = "Frequency"

      counter = 1
      For i = 1 To Round(Cells(5, 5).Value, 0) 'get bin values for band A
         Cells(7 + i, 1) = i
         Cells(7 + i, 2) = "=COUNTIF(" & dataSheet & "!A2:A" & numRows & ", ""=" & i & """)"
         counter = counter + 1
      Next i

      Cells(7 + counter, 1) = Cells(5, 6) 'put in next bin (band c end)
      Cells(1, 9).Formula = "=COUNTIF(" & dataSheet & "!A2:A" & numRows & ", ""<" & Cells(5,
6) & """)"
      Cells(1, 10).Formula = "=COUNTIF(" & dataSheet & "!A2:A" & numRows & ", ""<" &
Cells(4, 6) & """)"
      Cells(7 + counter, 2) = Abs(Cells(1, 9) - Cells(1, 10))
         counter = counter + 1

      Cells(7 + counter, 1) = Cells(5, 7) 'band b end
      Cells(1, 9).Formula = "=COUNTIF(" & dataSheet & "!A2:A" & numRows & ", ""<" & Cells(5,
7) & """)"
      Cells(1, 10).Formula = "=COUNTIF(" & dataSheet & "!A2:A" & numRows & ", ""<" &
Cells(4, 7) & """)"
      Cells(7 + counter, 2) = Abs(Cells(1, 9) - Cells(1, 10))
         counter = counter + 1

      exitIf = False
      If Cells(5, 7) < 15 Then         'add more bins 15-30...
         Cells(7 + counter, 1) = "15"
         Cells(1, 9).Formula = "=COUNTIF(" & dataSheet & "!A2:A" & numRows & ", ""<= 15"")"
         Cells(1, 10).Formula = "=COUNTIF(" & dataSheet & "!A2:A" & numRows & ", ""<" &
Cells(4, 8) & """)"
         Cells(7 + counter, 2) = Abs(Cells(1, 9) - Cells(1, 10))
         If Cells(7 + counter, 2) = 0 Then
            Cells(7 + counter, 1) = "> " & Cells(4, 8)
            Cells(7 + counter, 2).Formula = "=COUNTIF(" & dataSheet & "!A2:A" & numRows & ",
"">=" & Cells(4, 8) & """)"
               exitIf = True
            End If
```

```
                counter = counter + 1

            If exitIf = False Then
               Cells(7 + counter, 1) = "20"
               Cells(1, 9).Formula = "=COUNTIF(" & dataSheet & "!A2:A" & numRows & ", ""<=
20""")"
               Cells(1, 10).Formula = "=COUNTIF(" & dataSheet & "!A2:A" & numRows & ", ""<
16""")"
               Cells(7 + counter, 2) = Abs(Cells(1, 9) - Cells(1, 10))
               If Cells(7 + counter, 2) = 0 Then
                  Cells(7 + counter, 1) = "> 15"
                  Cells(7 + counter, 2).Formula = "=COUNTIF(" & dataSheet & "!A2:A" & numRows &
", "">= 15""")"
                  exitIf = True
               End If
            End If ' exitif
            counter = counter + 1

            If exitIf = False Then
               Cells(7 + counter, 1) = "25"
               Cells(1, 9).Formula = "=COUNTIF(" & dataSheet & "!A2:A" & numRows & ", ""<=
25""")"
               Cells(1, 10).Formula = "=COUNTIF(" & dataSheet & "!A2:A" & numRows & ", ""<
21""")"
               Cells(7 + counter, 2) = Abs(Cells(1, 9) - Cells(1, 10))
               If Cells(7 + counter, 2) = 0 Then
                  Cells(7 + counter, 1) = "> 20"
                  Cells(7 + counter, 2).Formula = "=COUNTIF(" & dataSheet & "!A2:A" & numRows &
", "">= 20""")"
                  exitIf = True
               End If
            End If ' exitif
            counter = counter + 1

            If exitIf = False Then
               Cells(7 + counter, 1) = "30"
               Cells(1, 9).Formula = "=COUNTIF(" & dataSheet & "!A2:A" & numRows & ", ""<=
30""")"
               Cells(1, 10).Formula = "=COUNTIF(" & dataSheet & "!A2:A" & numRows & ", ""<
26""")"
               Cells(7 + counter, 2) = Abs(Cells(1, 9) - Cells(1, 10))
               If Cells(7 + counter, 2) = 0 Then
                  Cells(7 + counter, 1) = "> 25"
                  Cells(7 + counter, 2).Formula = "=COUNTIF(" & dataSheet & "!A2:A" & numRows &
", "">= 25""")"
                  exitIf = True
               End If
            End If ' exitif
            counter = counter + 1
```

314

```
            If exitIf = False Then
               Cells(7 + counter, 1) = "> 30"
               Cells(7 + counter, 2) = "=COUNTIF(" & dataSheet & "!A2:A" & numRows & ", ""> 
30""")"
            End If

         Else
            If exitIf = False Then
               Cells(7 + counter, 1) = "20"
               Cells(1, 9).Formula = "=COUNTIF(" & dataSheet & "!A2:A" & numRows & ", ""<= 
20""")"
               Cells(1, 10).Formula = "=COUNTIF(" & dataSheet & "!A2:A" & numRows & ", ""<= 
15""")"
               Cells(7 + counter, 2) = Abs(Cells(1, 9) - Cells(1, 10))
               If Cells(7 + counter, 2) = 0 Then
                  Cells(7 + counter, 1) = "> 15"
                  Cells(7 + counter, 2).Formula = "=COUNTIF(" & dataSheet & "!A2:A" & numRows & 
", "">= 15""")"
                     exitIf = True
                  End If
               End If ' exitif
               counter = counter + 1

               If exitIf = False Then
                  Cells(7 + counter, 1) = "25"
                  Cells(1, 9).Formula = "=COUNTIF(" & dataSheet & "!A2:A" & numRows & ", ""<= 
25""")"
                  Cells(1, 10).Formula = "=COUNTIF(" & dataSheet & "!A2:A" & numRows & ", ""<= 
20""")"
                  Cells(7 + counter, 2) = Abs(Cells(1, 9) - Cells(1, 10))
                  If Cells(7 + counter, 2) = 0 Then
                     Cells(7 + counter, 1) = "> 20"
                     Cells(7 + counter, 2).Formula = "=COUNTIF(" & dataSheet & "!A2:A" & numRows & 
", "">= 20""")"
                        exitIf = True
                     End If
                  End If ' exitif
                  counter = counter + 1

                  If exitIf = False Then
                     Cells(7 + counter, 1) = "30"
                     Cells(1, 9).Formula = "=COUNTIF(" & dataSheet & "!A2:A" & numRows & ", ""<= 
30""")"
                     Cells(1, 10).Formula = "=COUNTIF(" & dataSheet & "!A2:A" & numRows & ", ""<= 
25""")"
                     Cells(7 + counter, 2) = Abs(Cells(1, 9) - Cells(1, 10))
                     If Cells(7 + counter, 2) = 0 Then
                        Cells(7 + counter, 1) = "> 25"
                        Cells(7 + counter, 2).Formula = "=COUNTIF(" & dataSheet & "!A2:A" & numRows & 
", "">= 25""")"
```

```
            exitIf = True
          End If
        End If ' exitif
        counter = counter + 1

        If exitIf = False Then
          Cells(7 + counter, 1) = "> 30"
          Cells(7 + counter, 2) = "=COUNTIF(" & dataSheet & "!A2:A" & numRows & ", "">
30"")"
        End If
      End If

    Call formatSheetForPrint

    End Sub
'------------------------------------------------------------------------
' Sub: CopyMathCadObj
' Author: Matt Behnke
' Created: 12/5/01
' Description: copies the mathcad onject, for running a curve fit..
' inputs:
'
' Outputs:
'------------------------------------------------------------------------
Sub CopyMathCadObj()

    Windows("AffiliationMacro.xls").Activate
    Sheets("Mathcad").Select
    Sheets("Mathcad").Copy Before:=Workbooks(currFilename).Sheets(dataSheet)
'   If ActiveSheet.Name = "Mathcad" Then
'       ActiveSheet.Name = "Mathcad_" & band
'   Else
'       MsgBox ("Mathcad sheet rename failed")
'   End If

    End Sub


'------------------------------------------------------------------------
' Sub: CopyDistributionGraph
' Author: Matt Behnke
' Created: 11/5/01
' Description: copies the distribution graph from the macro sheet into the data spreadsheet
' inputs:
'
' Outputs:
'------------------------------------------------------------------------
Sub CopyDistributionGraph()

Application.DisplayAlerts = False
```

316

```
        numSheets = Sheets.Count

        Windows("AffiliationMacro.xls").Activate
        Sheets("Affiliation Distribution Sample").Select
        Sheets("Affiliation                    Distribution                    Sample").Copy
After:=Workbooks(currFilename).Sheets(numSheets)
        ActiveChart.SeriesCollection(1).Select
        ActiveChart.SeriesCollection(1).XValues = "=Distribution!R8C1:R18C1"
        ActiveChart.SeriesCollection(1).Values = "=Distribution!R8C2:R19C2"
        ActiveChart.ChartTitle.Characters.Text = "Productivity Distribution" & Chr(10) _
        & technologyName & " (" & stepInterval & ")"

        Application.DisplayAlerts = True

        End Sub


        '-------------------------------------------------------------------------
        ' Sub: ComputeEntropy
        ' Author: Matt Behnke
        ' Created: 1/28/02
        ' Description: Computes the cumulative entropy using the supplied datasheets
        '          note number of instances must begin at row 2, column 4..
        ' inputs: datasheet - matrix of the descriptorData.. Y-axis is the terms, X-axis is timesteps, v is #
of instances
        '          time  1,  2,  3,  4 .....
        '          term1 v   v   v
        '          term2 v
        '        outSheet: name of the sheet that contains the computed entropy.
        '        theType: 1) s(x|y), 2) s(y|x)
        ' Outputs:
        '-------------------------------------------------------------------------
        Sub ComputeEntropy(ByVal dataSheet As String, ByVal outSheet As String, ByVal theType As
Integer)

          numCols = CountCols(dataSheet, 1)
          numRows = CountRows(dataSheet, 1)

          Sheets.Add After:=Worksheets(Worksheets.Count)
          Sheets(Sheets.Count).Select
          ActiveSheet.Name = outSheet
          Worksheets(outSheet).Move After:=Worksheets(dataSheet)


          For i = 1 To numCols
            If i >= 4 And theType = 1 Then
              TotalNumInstances = Sheets(dataSheet).Cells(numRows + 1, i)
            ElseIf i >= 4 And theType = 2 Then
              TotalNumInstances = Sheets(dataSheet).Cells(numRows + 1, 4)
            End If
```

```vbnet
        For j = 1 To numRows

            If i >= 4 And j >= 2 Then
                numInstances = Sheets(dataSheet).Cells(j, i)
                If numInstances > 0 Then
                    entropy   =   -numInstances   /   TotalNumInstances   *   (Log(numInstances   /
TotalNumInstances) / Log(2))
                    Sheets(outSheet).Cells(j, i) = entropy
                End If
                If j = numRows Then 'put in sum of entropy
                    Sheets(outSheet).Cells(j + 1, i) = "=SUM(" & col(i) & "2:" & col(i) & numRows & ")"
                End If
            Else 'copy terms, count, first pub date
                Sheets(outSheet).Cells(j, i).Value = Sheets(dataSheet).Cells(j, i).Value
            End If
        Next j
    Next i

End Sub

'---------------------------------------------------------------------------
' Sub: CreateDescriptorDataY
' Author: Matt Behnke
' Created: 1/28/02
' Description: Takes the supplied descriptor data sheet and creates the Y part of the (X,Y) world
'              as x increases y decreases.. a value decreases on the y sheet when a value increases on
the y sheet
' inputs: datasheet - matrix of the descriptorData.. Y-axis is the terms, X-axis is timesteps, v is #
of instances
'         time  1,  2,  3,  4 .....
'         term1 v  v  v
'         term2 v
'       outSheet: name of the sheet that contains DescriptorDataY
' Outputs:
'---------------------------------------------------------------------------

Sub CreateDescriptorDataY(ByVal dataSheet As String, ByVal outSheet As String)

    numCols = CountCols(dataSheet, 1)
    numRows = CountRows(dataSheet, 1)

    Worksheets(dataSheet).Copy After:=Worksheets(dataSheet)

    Sheets(dataSheet & " (2)").Select
    ActiveSheet.Name = outSheet

    For i = 2 To numRows

        For j = 4 To numCols
```

```vb
            numTotalInstances = Sheets(outSheet).Cells(i, 2)
            numInstances = Sheets(outSheet).Cells(i, j)

            If j = 4 And Sheets(outSheet).Cells(i, j) > 0 Then 'places the initial value at the end..
               lastColumn = Sheets(outSheet).Cells(i, j)
            End If

            Sheets(outSheet).Cells(i, j) = numTotalInstances - numInstances

            If j = numCols And lastColumn > 0 Then
               Sheets(outSheet).Cells(i, j) = lastColumn 'places the value of first column x into last
coln Y.
            End If

            If i = numRows Then 'put in sum
               Sheets(outSheet).Cells(i + 1, j) = "=SUM(" & col(j) & "2:" & col(j) & numRows & ")"
            End If
         Next j
         lastColumn = 0
      Next i

End Sub 'CreateDescriptorDataY


Sub computeEntropyTest()
'IT WORKS

'Call ComputeEntropy("descriptor_data_X", "World_Cumulative_Entropy_X", 1)
Call ComputeEntropy("descriptor_data_Y", "World_Cumulative_Entropy_Y", 2)
'Call CreateDescriptorDataY("descriptor_data_X", "descriptor_data_Y")
End Sub

'---------------------------------------------------------------------------
'  Sub: FillBand
'  Author: Matt Behnke
'  Created: 11/7/01
'  Description: fills in a bands distribution by copying a row from the list of all the affiliations
(datasheet)
'  inputs: band name
'
'  Outputs:
'---------------------------------------------------------------------------
Sub FillBand(ByVal band As String)

   Sheets.Add After:=Worksheets(Worksheets.Count)

   numRows = CountRows(dataSheet, 1)
   Sheets(Worksheets.Count).Select
   currSheetName = ActiveSheet.Name
   Columns("C:C").ColumnWidth = 62.43
```

```
     Select Case band
        Case "A_Band"
           bandFrom = Sheets("Distribution").Cells(4, 8)
           bandTo = 32500
        Case "B_Band"
           bandFrom = Sheets("Distribution").Cells(4, 7)
           bandTo = Sheets("Distribution").Cells(5, 7)
        Case "C_Band"
           bandFrom = Sheets("Distribution").Cells(4, 6)
           bandTo = Sheets("Distribution").Cells(5, 6)
        Case "D_Band"
           bandFrom = Sheets("Distribution").Cells(4, 5)
           bandTo = Sheets("Distribution").Cells(5, 5)
     End Select

     Sheets("" & dataSheet & "").Select
     Rows("1:1").Select
     Selection.Copy
     Sheets(currSheetName).Select
     Rows("1:1").Select
     ActiveSheet.Paste

     counter = 2
     For i = 2 To numRows 'copy rows from datasheet into band
        If Sheets(dataSheet).Cells(i, 1) >= bandFrom And Sheets(dataSheet).Cells(i, 1) <= bandTo
Then
           Sheets(dataSheet).Select
           Rows(i & ":" & i).Select
           Selection.Copy
           Sheets(currSheetName).Select
           Rows(counter & ":" & counter).Select
           ActiveSheet.Paste
           counter = counter + 1
        End If
     Next i

     numRowsInBand = CountRows(currSheetName, 1)
     numColumns = CountCols(currSheetName, 1) 'num time steps

     Cells(numRowsInBand + 1, 3) = "Count"
     Cells(numRowsInBand + 2, 3) = "Mean"
     Cells(numRowsInBand + 3, 3) = "Std Dev"
     Cells(numRowsInBand + 4, 3) = "Sum"

     For i = 4 To numColumns 'put in the mean and std deviation for each time step
'put in zeros if nothing there
'        For j = 2 To numRowsInBand
'           If i = 4 Then
'              If Cells(j, i) > 0 Then
```

320

```
'            Else
'                Cells(j, i) = 0
'            End If
'        Else
'            Cells(j, i) = Cells(j, i) + Cells(j, i - 1)
'        End If
'    Next j

'dont put in zeros if nothing there
'    For j = 2 To numRowsInBand
'        If (Cells(j, i) > 0 And i > 4) Or (i > 4 And Cells(j, i - 1) > 0) Then
'            Cells(j, i) = Cells(j, i) + Cells(j, i - 1)
'        End If
'    Next j

        Cells(numRowsInBand + 4, i).Formula = "=Sum(" & col(i) & "2:" & col(i) &
numRowsInBand & ")"
        Cells(numRowsInBand + 1, i).Formula = "=Countif(" & col(i) & "2:" & col(i) &
numRowsInBand & ", "">0"")"
        If Cells(numRowsInBand + 1, i) > 0 Then
          Cells(numRowsInBand + 2, i).Formula = "=AVERAGE(" & col(i) & "2:" & col(i) &
numRowsInBand & ")"
          If Cells(numRowsInBand + 1, i) > 1 Then 'more than one so comput std deviation
            Cells(numRowsInBand + 3, i).Formula = "=STDEV(" & col(i) & "2:" & col(i) &
numRowsInBand & ")"
          End If
        End If
      Next i

      ActiveSheet.Name = "Affiliation_Cum_Dist_" & band
      Call formatSheetForPrint

End Sub

'---------------------------------------------------------------------------
' Sub: FillBandStats
' Author: Matt Behnke
' Created: 11/7/01
' revised: 12/3/01
' Description: creates a band's statistics sheet
' inputs: band name
'
' Outputs:
'---------------------------------------------------------------------------
Sub FillBandStats(ByVal band As String)

  Dim data As Variant

  Sheets.Add After:=Worksheets(Worksheets.Count)
```

```vba
If band = "World" Then
    source = dataSheet
Else
    source = "Affiliation_Cum_Dist_" & band
End If

numRowsInBand = CountRows(source, 1)
numTimeStepsInBand = CountCols(source, 1) - 3
Sheets(Worksheets.Count).Select
'Columns("C:C").ColumnWidth = 62.43

Cells(5, 1) = " "
Cells(6, 1) = " "
Cells(7, 1) = " "
Cells(8, 1) = " "
Cells(9, 1) = " "
Cells(11, 1) = " "
Cells(12, 1) = " "


Cells(1, 1) = "Curve fit y(t)  y(t) = bt^m"
Cells(3, 1) = "b"
Cells(4, 1) = "m"

Cells(8, 3) = "Total Production"
Cells(8, 6) = "Production/Step (on Average)"
Cells(8, 11) = "Calculated Production/Step)"

Cells(10, 1) = "Time Step"
Cells(10, 2) = "Step Name"

Cells(10, 3) = "Mean"
Cells(10, 4) = "Std Deviation"
Cells(10, 5) = "'+ 3 sigma"
'average per step
Cells(10, 6) = "Mean"
Cells(10, 7) = "Std Deviation"
Cells(10, 8) = "'+ 3 sigma"

Cells(10, 9) = "Total Prod"
Cells(10, 10) = "kappa"
Cells(10, 11) = "kappa/2"
Cells(10, 12) = "r value"

Cells(10, 13) = "Mean"
Cells(10, 14) = "R^2"
Cells(10, 15) = "'+ 3 sigma"


Cells(13, 1) = "0" 'time step zero
```

```
For i = 1 To numTimeStepsInBand
   Cells(13 + i, 1) = i 'step number
   Cells(13 + i, 2) = Sheets(source).Cells(1, i + 3) 'step name
  'Total Production
   Cells(13 + i, 3) = Sheets(source).Cells(numRowsInBand + 2, i + 3) 'mean
   Cells(13 + i, 4) = Sheets(source).Cells(numRowsInBand + 3, i + 3) 'stdev
   Cells(13 + i, 5) = Cells(13 + i, 3) + 3 * Cells(13 + i, 4) 'mean + 3std
  'Production per step on avg..
   If i = 1 Then
      'Cells(13 + i, 6) = Cells(13 + i, 3) / Cells(13 + i, 1) 'mean
      Cells(13 + i, 6) = Cells(13 + i, 3)
      Cells(13 + i, 7) = Cells(13 + i, 4) 'stdev
      Cells(13 + i, 8) = Cells(13 + i, 5) 'mean * 3std
   Else
      Cells(13 + i, 6) = Cells(13 + i, 3) - Cells(12 + i, 3)
      Cells(13 + i, 7) = Cells(13 + i, 4) - Cells(12 + i, 4) 'stdev
      Cells(13 + i, 8) = Cells(13 + i, 5) - Cells(12 + i, 5) 'mean + 3std
   End If

   If i = numTimeStepsInBand Then  'put in average
      Cells(14 + i, 8) = "=AVERAGE(H14:H" & i + 13 & ")"
   End If

   If i = 1 Then
      Cells(13 + i, 9) = Cells(13 + i, 8)
   Else
      Cells(13 + i, 9) = Cells(13 + i, 8) + Cells(12 + i, 9)
   End If

   Cells(13 + i, 10) = "=H" & 14 + numTimeStepsInBand 'avg of mean*3std
Next i

ActiveSheet.Name = "" & band & "_Stats"
Call copyStatGraphs(numTimeStepsInBand, band, "" & band & "_Stats")

Sheets("" & band & "_Stats").Select

'get formula of trendline from entropy power trend graph
trendEq = Cells(2, 1)
Cells(3, 2) = firstPartTrendEq(trendEq)
Cells(4, 2) = secondPartTrendEq(trendEq)

'get kappa, r. p

Cells(3, 3) = "kappa"  'headers
Cells(4, 3) = "r"
Cells(5, 3) = "p"
Cells(6, 3) = "1-Sum(r^2)"
```

```
        j = 14 'get the start row
        While Not (i > 0#)   'if the first time step's mean is zero find the step that doesnt have 0
          i = Cells(j, 3).Value
          If Not (i > 0) Then
            j = j + 1
          End If
        Wend

        'j = j + 1 'add one to the starting row to not include the first time step....
        numRowsToUse = numTimeStepsInBand - (j - 14)
        data = Update_Mathcad_Band_Stats("Mathcad", ActiveSheet.Name, "C" & j, "F" & j,
numRowsToUse, 0, 0.001)

        kappa = data(1)
        r = data(2)
        p = data(3)
        r2a = data(4)

        'put on sheet
        Cells(3, 4) = Round(kappa, 4)
        Cells(4, 4) = Round(r, 4)
        Cells(5, 4) = Round(p, 4)
        Cells(6, 5) = Round(r2a, 4)

      'calculate prediticded means for -1, -2 under total
       'Cells(11, 3).Formula = "=-$B$3*-A11^$B$4"   'not needed
       'Cells(12, 3).Formula = "=-$B$3*-A12^$B$4"
       Cells(13, 3) = 0


        For i = 1 To numTimeStepsInBand
          'fill in kappa, kappa/2, rvalue
          Cells(13 + i, 10) = "=$D$3"
          Cells(13 + i, 11) = "=$D$3 / 2"
          Cells(13 + i, 12) = "=$D$4"

        'fill in calculated prod per step
          Cells(13 + i, 13).Formula = "=$D$3*(C" & 13 + i & "+$D$5)/(C" & 13 + i &
"+$D$4+$D$5)" 'mean
          Cells(13 + i, 14) = "=(M" & 13 + i & "-F" & 13 + i & ")*(M" & 13 + i & "-F" & 13 + i & ")"
'R^2
          sumRSquared = Cells(13 + i, 14) + sumRSquared
          Cells(13 + i, 15) = "=$D$3*(E" & 13 + i & "+$D$5)/(E" & 13 + i & "+$D$4+$D$5)"

          If i = numTimeStepsInBand Then   'put in average R^2 -------- REMOVE AFTER dbl
Checking values......
              Cells(15 + i, 10) = "Sum(R^2)"
              Cells(15 + i, 12) = "=Sum(N14:N" & i + 13 & ")"
              Cells(16 + i, 12) = "=1-L" & i + 15
          End If
```

```
        Next i

        inverseRSquared = (1 - sumRSquared) 'from the sum of r squared
        Cells(6, 4) = Round(inverseRSquared, 4) '4decimal places

    '   Cells(11, 11).Formula = "=$D$3*(C11+$D$5)/(C11+$D$4+$D$5)"  -removed (-2, -1, 0 time
steps of calculated mean)
    '   Cells(12, 11).Formula = "=$D$3*(C12+$D$5)/(C12+$D$4+$D$5)"
    '   Cells(13, 11) = "=$D$3*(C13+$D$5)/(C13+$D$4+$D$5)"

        Call formatSheetForPrint
        Call copyLearningCum(numTimeStepsInBand, band, "" & band & "_Stats") 'learning vs. cum

    End Sub 'fill stats

    '--------------------------------------------------------------------------
    ' Sub: copyStatGraphs
    ' Author: Matt Behnke
    ' Created: 11/7/01
    ' Description: copyies the affilaition statistics graphs
    ' inputs:
    '
    ' Outputs:
    '--------------------------------------------------------------------------
    Sub copyStatGraphs(ByVal timeSteps As Integer, ByVal band As String, ByVal source As String)

    Application.DisplayAlerts = False

        numSheets = Sheets.Count

        Windows("AffiliationMacro.xls").Activate
        Sheets("A_Band_Learning_Cap_per_k").Select
        Sheets("A_Band_Learning_Cap_per_k").Copy
After:=Workbooks(currFilename).Sheets(numSheets)
        ActiveChart.SeriesCollection(1).Select

        j = 14
        While Not (i > 0#)   'if the first time step's mean is zero find the step that doesnt have 0
          i = Sheets(source).Cells(j, 3).Value
          If Not (i > 0) Then
            j = j + 1
          End If
        Wend

        ActiveChart.SeriesCollection(1).Values = "=" & source & "!R" & j & "C6:R" & timeSteps + 13
& "C6"

        ActiveChart.SeriesCollection(2).Values = "=" & source & "!R" & j + 1 & "C8:R" & timeSteps
+ 13 & "C8"
```

```
        ActiveChart.ChartTitle.Characters.Text = "" & band & " Productivity Index (Cum over k)" &
Chr(10) _
        & technologyName & " (" & stepInterval & ")"
        ActiveSheet.Name = "" & band & "_Learning_Cap_per_k"

        ActiveChart.SeriesCollection(1).ErrorBars.Select
        ExecuteExcel4Macro _
          "ERRORBAR.Y(2,5,""="  &  source  &  "!R"  &  j  &  "C7:R"  &  timeSteps  +  13  &
"C7"",""=A_Band_Stats!$F$" & j & ":$F$" & timeSteps + 13 & """)"

      'move legend and textbox
       ActiveChart.Legend.Select
       Selection.Left = 431
       Selection.Top = 341

      'copy second graph
       Windows("AffiliationMacro.xls").Activate
       Sheets("A_Band_Learning_Cum").Select
       Sheets("A_Band_Learning_Cum").Copy After:=Workbooks(currFilename).Sheets(numSheets)
       ActiveChart.SeriesCollection(1).Select

       ActiveChart.SeriesCollection(1).XValues = "=" & source & "!R" & j & "C1:R" & timeSteps +
13 & "C1"
       ActiveChart.SeriesCollection(1).Values = "=" & source & "!R" & j & "C3:R" & timeSteps + 13
& "C3"

       ActiveChart.SeriesCollection(2).XValues = "=" & source & "!R" & j & "C1:R" & timeSteps +
13 & "C1"
       ActiveChart.SeriesCollection(2).Values = "=" & source & "!R" & j & "C5:R" & timeSteps + 13
& "C5"

       ActiveChart.SeriesCollection(1).Select
      With ActiveChart.SeriesCollection(1).Trendlines(1)
       'put trendline equation onto stats sheet
        Worksheets(source).Cells(2, 1).Value = .DataLabel.Text
       .DisplayRSquared = True
      End With

        ActiveChart.ChartTitle.Characters.Text = "" & band & " Productivity In Pubs (Cum over k)" &
Chr(10) _
        & technologyName & " (" & stepInterval & ")"
        ActiveSheet.Name = "" & band & "_Learning_Cum"

      Application.DisplayAlerts = True

      End Sub
```

```
'**********
' copies the learning vs cumulative chart..........
'
'


        Sub copyLearningCum(ByVal timeSteps As Integer, ByVal band As String, ByVal source As
String)

        Application.DisplayAlerts = False

          kappa = Sheets(source).Cells(3, 4)
          r = Sheets(source).Cells(4, 4)
          p = Sheets(source).Cells(5, 4)
          r2 = Sheets(source).Cells(6, 4)

         numSheets = Sheets.Count

        Windows("AffiliationMacro.xls").Activate
          Sheets("A_Band_Learning_Vs_Cum").Select
          Sheets("A_Band_Learning_Vs_Cum").Copy After:=Workbooks(currFilename _
            ).Sheets(numSheets)
          ActiveChart.PlotArea.Select

          j = 14
          While Not (i > 0#)   'if the first time step's mean is zero find the step that doesnt have 0
            i = Sheets(source).Cells(j, 3).Value
            If Not (i > 0) Then
              j = j + 1
            End If
          Wend


        ActiveChart.SeriesCollection(1).XValues = "=" & source & "!R" & j & "C3:R" & timeSteps +
13 & "C3"
        ActiveChart.SeriesCollection(1).Values = "=" & source & "!R" & j & "C6:R" & timeSteps + 13
& "C6"

        ActiveChart.SeriesCollection(2).XValues = "=" & source & "!R" & j & "C3:R" & timeSteps +
13 & "C3"
        ActiveChart.SeriesCollection(2).Values = "=" & source & "!R14C13:R" & timeSteps + 13 &
"C13"

        'kappa
        ActiveChart.SeriesCollection(3).XValues = "=" & source & "!R" & j & "C5:R" & timeSteps +
13 & "C5"
        ActiveChart.SeriesCollection(3).Values = "=" & source & "!R14C10:R" & timeSteps + 13 &
"C10"

        '3sigma 3sigma
```

```vba
        ActiveChart.SeriesCollection(4).XValues = "=" & source & "!R" & j & "C5:R" & timeSteps +
13 & "C5" 'E
        ActiveChart.SeriesCollection(4).Values = "=" & source & "!R14C8:R" & timeSteps + 13 &
"C8" 'H


        'kappa/2
        ActiveChart.SeriesCollection(5).XValues = "=" & source & "!R" & j & "C5:R" & timeSteps +
13 & "C5"
        ActiveChart.SeriesCollection(5).Values = "=" & source & "!R14C11:R" & timeSteps + 13 &
"C11"


        'r-p
        ActiveChart.SeriesCollection(6).XValues = "=" & source & "!R" & j & "C12:R" & timeSteps +
13 & "C12"
        ActiveChart.SeriesCollection(6).Values = "=" & source & "!R14C6:R" & timeSteps + 13 &
"C6"

    '   ActiveChart.SeriesCollection(1).ErrorBars.Select
    '   ExecuteExcel4Macro _
    '        "ERRORBAR.Y(2,5,""=" & source & "!R" & j & "C7:R" & timeSteps + 13 &
"C7"",""=A_Band_Stats!$F$" & j & ":$F$" & timeSteps + 13 & """)"

        ActiveChart.Shapes("Text Box 6").Select
        Selection.Characters.Text = "K= " & kappa & Chr(10) & "r= " & r & Chr(10) & "p= " & p &
Chr(10) & "" & Chr(10) & "R2= " & r2 & Chr(10) & "" & Chr(10) & ""

        ActiveChart.ChartTitle.Characters.Text = "Learning Curve -- " & band & " (Mean and
Capacity)" & Chr(10) _
        & technologyName & " (" & stepInterval & ")"
        ActiveSheet.Name = "" & band & "_Learning_Vs_Cum"
    Application.DisplayAlerts = True

    End Sub
    '-------------------------------------------------------------------------
    ' Sub: CopyABCDGraph
    ' Author: Matt Behnke
    ' Created: 11/7/01
    ' Description: copies the ABCD band mean graph and changes the dataseries to point to the right
data..
    ' inputs:
    '
    ' Outputs:
    '-------------------------------------------------------------------------
    Sub CopyABCDGraph()

    Application.DisplayAlerts = False

        kappa = Sheets("A_Band_Stats").Cells(3, 4)
        r = Sheets("A_Band_Stats").Cells(4, 4)
        p = Sheets("A_Band_Stats").Cells(5, 4)
```

```vb
        r2 = Sheets("A_Band_Stats").Cells(6, 4)

        Windows("AffiliationMacro.xls").Activate
        Sheets("ABCD_Band_Learning_Vs_Cum").Select
        Sheets("ABCD_Band_Learning_Vs_Cum").Copy After:=Workbooks( _
          currFilename).Sheets(Sheets.Count)
        ActiveChart.PlotArea.Select
    '   ActiveChart.SeriesCollection(3).Delete
        currChartName = ActiveChart.Name

        timeSteps = CountCols("Affiliation_Cum_Dist_A_Band", 1) - 3
     'aband
       j = 14
       While Not (i > 0#)   'if the first time step's mean is zero find the step that doesnt have 0
         i = Sheets("A_Band_Stats").Cells(j, 3).Value
         If Not (i > 0) Then
            j = j + 1
         End If
       Wend

        Charts(currChartName).Select
        'aband mean
        ActiveChart.SeriesCollection(1).XValues = "=A_Band_Stats!R" & j & "C3:R" & timeSteps +
13 & "C3"
        ActiveChart.SeriesCollection(1).Values = "=A_Band_Stats!R" & j & "C6:R" & timeSteps + 13
& "C6"

        'calc y
        ActiveChart.SeriesCollection(2).XValues = "=A_Band_Stats!R" & j + 1 & "C3:R" & timeSteps
+ 13 & "C3"
        ActiveChart.SeriesCollection(2).Values = "=A_Band_Stats!R" & j + 1 & "C13:R" & timeSteps
+ 13 & "C13"

        '3sigma 3sigma
        ActiveChart.SeriesCollection(3).XValues = "=A_Band_Stats!R" & j + 1 & "C5:R" & timeSteps
+ 13 & "C5"
        ActiveChart.SeriesCollection(3).Values = "=A_Band_Stats!R" & j & "C15:R" & timeSteps +
13 & "C15"

        'aband kappa
        ActiveChart.SeriesCollection(7).XValues = "=A_Band_Stats!R" & j & "C5:R" & timeSteps +
13 & "C5"
        ActiveChart.SeriesCollection(7).Values = "=A_Band_Stats!R" & j & "C10:R" & timeSteps +
13 & "C10"

        'aband 3sig 3sig
        ActiveChart.SeriesCollection(8).XValues = "=A_Band_Stats!R" & j & "C5:R" & timeSteps +
13 & "C5"
        ActiveChart.SeriesCollection(8).Values = "=A_Band_Stats!R" & timeSteps + 14 & "C8"
```

```vba
        'aband kappa /2
        ActiveChart.SeriesCollection(9).XValues = "=A_Band_Stats!R" & j & "C5:R" & timeSteps +
13 & "C5"
        ActiveChart.SeriesCollection(9).Values = "=A_Band_Stats!R14C11:R" & timeSteps + 13 &
"C11"

        'aband r-p
        ActiveChart.SeriesCollection(10).XValues = "=A_Band_Stats!R" & j & "C12:R" & timeSteps
+ 13 & "C12"
        ActiveChart.SeriesCollection(10).Values = "=A_Band_Stats!R14C6:R" & timeSteps + 13 &
"C6"



        'ActiveChart.SeriesCollection(1).ErrorBars.Select
        'ExecuteExcel4Macro _
        '       "ERRORBAR.Y(2,5,""=A_Band_Stats!R" & j & "C7:R" & timeSteps + 13 &
"C7"",""=A_Band_Stats!$F$" & j & ":$F$" & timeSteps + 13 & """)"

      'bband **********THIS IS CORRECT.............
        j = 14
        While Not (i > 0#)   'if the first time step's mean is zero find the step that doesnt have 0
          i = Sheets("B_Band_Stats").Cells(j, 3).Value
          If Not (i > 0) Then
            j = j + 1
          End If
        Wend

        ActiveChart.SeriesCollection(4).XValues = "=B_Band_Stats!R" & j & "C3:R" & timeSteps +
13 & "C3"
        ActiveChart.SeriesCollection(4).Values = "=B_Band_Stats!R" & j & "C6:R" & timeSteps + 13
& "C6"

      'cband
        j = 14
        While Not (i > 0#)   'if the first time step's mean is zero find the step that doesnt have 0
          i = Sheets("C_Band_Stats").Cells(j, 3).Value
          If Not (i > 0) Then
            j = j + 1
          End If
        Wend

        ActiveChart.SeriesCollection(5).XValues = "=C_Band_Stats!R" & j & "C3:R" & timeSteps +
13 & "C3"
        ActiveChart.SeriesCollection(5).Values = "=C_Band_Stats!R" & j & "C6:R" & timeSteps + 13
& "C6"

      'dband
        j = 14
        While Not (i > 0#)   'if the first time step's mean is zero find the step that doesnt have 0
```

```vba
        i = Sheets("D_Band_Stats").Cells(j, 3).Value
        If Not (i > 0) Then
            j = j + 1
        End If
    Wend

    ActiveChart.SeriesCollection(6).XValues = "=D_Band_Stats!R" & j & "C3:R" & timeSteps + 13 & "C3"
    ActiveChart.SeriesCollection(6).Values = "=D_Band_Stats!R" & j & "C6:R" & timeSteps + 13 & "C6"

    'kappa textbox
    ActiveChart.Shapes("Text Box 7").Select
    Selection.Characters.Text = "K= " & kappa & Chr(10) & "r= " & r & Chr(10) & "p= " & p & Chr(10) & "" & Chr(10) & "R2= " & r2 & Chr(10) & "" & Chr(10) & ""

    ActiveChart.ChartTitle.Characters.Text = ActiveChart.ChartTitle.Characters.Text & Chr(10) _
    & technologyName & " (" & stepInterval & ")"

    Application.DisplayAlerts = True

End Sub

'-------------------------------------------------------------------------
' Sub: copyBandSummaryGraphs
' Author: Matt Behnke
' Created: 12/13/01
' Description: Copies the band ENTROPY graphs and published messages summary graphs
' inputs: band name
'
' Outputs:
'-------------------------------------------------------------------------

Sub CopyBandSummaryGraphs(ByVal band As String)

Application.DisplayAlerts = False

    If band = "World" Then
        source = "Affiliation_Summary"
    Else
        source = "Affiliation_Summary_" & band
    End If

    numRows = CountRows(source, 1)

'GRAPH ONE message_N_k+1 vs N_k

    Windows("AffiliationMacro.xls").Activate
    Sheets("A_Band_Message_N_k+1 vs N_k").Select
```

```
        Sheets("A_Band_Message_N_k+1                    vs                    N_k").Copy
Before:=Workbooks(currFilename).Sheets(band & "_Stats")
        ActiveChart.SeriesCollection(1).Select

        j = 4
        While Not (i > 0#)   'if the first time step's mean is zero find the step that doesnt have 0
           i = Sheets(source).Cells(j, 3).Value
           If Not (i > 0) Then
              j = j + 1
           End If
        Wend

        If  Sheets(source).Cells(i,  2).Characters(1,  1).Text  =  "1"  And  Sheets(source).Cells(i,
2).Characters(2, 1).Text = "/" Then
           yx = j
        Else
           yx = j - 1
        End If

        ActiveChart.SeriesCollection(1).XValues = "=" & source & "!R" & j & "C3:R" & numRows - 1
& "C3"
        ActiveChart.SeriesCollection(1).Values = "=" & source & "!R" & j + 1 & "C3:R" & numRows
& "C3"
        'y=x:
        ActiveChart.SeriesCollection(2).XValues = "=" & source & "!R" & yx & "C3:R" & numRows
& "C3"
        ActiveChart.SeriesCollection(2).Values = "=" & source & "!R" & yx & "C3:R" & numRows &
"C3"

        titleBefore = ActiveChart.ChartTitle.Characters.Text

        ActiveChart.ChartTitle.Characters.Text = band & " " & titleBefore & Chr(10) _
        & technologyName & " (" & stepInterval & ")"

        'place subscripts in the chart title (N_k+1, N_k)
        If band = "World" Then
           ActiveChart.ChartTitle.Select
           With Selection.Characters(Start:=29, Length:=3).Font
              .Subscript = True
           End With
           With Selection.Characters(Start:=34, Length:=1).Font
              .Subscript = True
           End With
        Else
           ActiveChart.ChartTitle.Select
           With Selection.Characters(Start:=30, Length:=3).Font
              .Subscript = True
           End With
           With Selection.Characters(Start:=35, Length:=1).Font
              .Subscript = True
```

```
            End With
          End If

          ActiveSheet.Name = "" & band & "_Message_N_k+1 vs N_k"

        'copy second graph S_k+1 vs S_k
          Windows("AffiliationMacro.xls").Activate
          Sheets("A_Band_World_S_k+1 vs S_k").Select
          Sheets("A_Band_World_S_k+1 vs S_k").Copy Before:=Workbooks(currFilename).Sheets(band
& "_Stats")
          ActiveChart.SeriesCollection(1).Select

          ActiveChart.SeriesCollection(1).XValues = "=" & source & "!R" & j & "C6:R" & numRows - 1
& "C6"
          ActiveChart.SeriesCollection(1).Values = "=" & source & "!R" & j + 1 & "C6:R" & numRows
& "C6"
          'y=x:
          ActiveChart.SeriesCollection(2).XValues = "=" & source & "!R" & yx & "C6:R" & numRows
& "C6"
          ActiveChart.SeriesCollection(2).Values = "=" & source & "!R" & yx & "C6:R" & numRows &
"C6"


          titleBefore = ActiveChart.ChartTitle.Characters.Text

          ActiveChart.ChartTitle.Characters.Text = band & " " & titleBefore & Chr(10) _
          & technologyName & " (" & stepInterval & ")"

          ActiveSheet.Name = "" & band & "_Entropy_S_k+1 vs S_k"

          'place subscripts in the chart title (Entropy S_k+1, S_k)
          If band = "World" Then
            ActiveChart.ChartTitle.Select
            With Selection.Characters(Start:=24, Length:=3).Font
              .Subscript = True
            End With
            With Selection.Characters(Start:=34, Length:=1).Font
              .Subscript = True
            End With
          Else
            ActiveChart.ChartTitle.Select
            With Selection.Characters(Start:=25, Length:=3).Font
              .Subscript = True
            End With
            With Selection.Characters(Start:=35, Length:=1).Font
              .Subscript = True
            End With
          End If
```

333

```
        If band = "World" Then

        Else
    'copy third Graph S(Y)_k+1 vs S_world_k
        Windows("AffiliationMacro.xls").Activate
        Sheets("A_Band_S(Y)_k+1 Vs S_world_k").Select
        Sheets("A_Band_S(Y)_k+1              Vs              S_world_k").Copy
Before:=Workbooks(currFilename).Sheets(band & "_Stats")
        ActiveChart.SeriesCollection(1).Select

        ActiveChart.SeriesCollection(1).XValues = "=Affiliation_Summary!R" & j & "C6:R" &
numRows & "C6"
        ActiveChart.SeriesCollection(1).Values = "=" & source & "!R" & j + 1 & "C6:R" &
numRows & "C6"
            'y=x
        ActiveChart.SeriesCollection(2).XValues = "=Affiliation_Summary!R" & yx & "C6:R" &
numRows & "C6"
        ActiveChart.SeriesCollection(2).Values = "=Affiliation_Summary!R" & yx & "C6:R" &
numRows & "C6"

        titleBefore = ActiveChart.ChartTitle.Characters.Text

        ActiveChart.ChartTitle.Characters.Text = band & " " & titleBefore & Chr(10) _
        & technologyName & " (" & stepInterval & ")"

        'subscripts in chart title
        ActiveChart.ChartTitle.Select
          With Selection.Characters(Start:=24, Length:=4).Font
            .Subscript = True
          End With
          With Selection.Characters(Start:=33, Length:=3).Font
            .Subscript = True
          End With
          With Selection.Characters(Start:=39, Length:=5).Font
            .Subscript = True
          End With
          With Selection.Characters(Start:=49, Length:=1).Font
            .Subscript = True
          End With
        ActiveSheet.Name = "" & band & "_S(X,Y)_k+1 vs S_world_k"
      End If

    Application.DisplayAlerts = True

    End Sub 'copy summary band graphs


    '-------------------------------------------------------------------------
    ' Sub: FillBandAuthors
    ' Author: Matt Behnke
    ' Created: 11/7/01
```

```vba
'  Description: fills in a bands author distribution by copying a row from the list of
'               affilations with the number of authors as the matrix's values.
'  inputs: band name
'
'  Outputs:
'------------------------------------------------------------------------
Sub FillBandAuthors(ByVal band As String)

    Sheets.Add After:=Worksheets(Worksheets.Count)

    numRowsInBand = CountRows("Affiliation_Cum_Dist_" & band, 1)
    numRowsInAuthors = CountRows("Affiliation_Authors", 1)

    Sheets(Worksheets.Count).Select
    currSheetName = ActiveSheet.Name
    Columns("C:C").ColumnWidth = 62.43

    Sheets(currSheetName).Move Before:=Sheets("" & band & "_Stats")

    Sheets("Affiliation_Authors").Select
    Rows("1:1").Select
    Selection.Copy
    Sheets(currSheetName).Select
    Rows("1:1").Select
    ActiveSheet.Paste

    counter = 2
    For i = 2 To numRowsInBand 'copy rows from datasheet into band
        affiliationName = Sheets("Affiliation_Cum_Dist_" & band).Cells(i, 3).Value
        For j = 2 To numRowsInAuthors
            If Sheets("Affiliation_Authors").Cells(j, 3).Value = affiliationName Then
                Sheets("Affiliation_Authors").Select
                Rows(j & ":" & j).Select
                Selection.Copy
                Sheets(currSheetName).Select
                Rows(counter & ":" & counter).Select
                ActiveSheet.Paste
                counter = counter + 1
            End If
        Next j
    Next i

    numRowsInAuthorBand = CountRows(currSheetName, 1)
    numColumns = CountCols(currSheetName, 1) 'num time steps

    Cells(numRowsInAuthorBand + 1, 3) = "Count"
    Cells(numRowsInAuthorBand + 2, 3) = "Mean"
    Cells(numRowsInAuthorBand + 3, 3) = "Std Dev"
    Cells(numRowsInAuthorBand + 4, 3) = "Sum"
```

```vba
        For i = 4 To numColumns 'put in the mean and std deviation for each time step

'        For j = 2 To numRowsInAuthorBand
'            If (Cells(j, i) > 0 And i > 4) Or (i > 4 And Cells(j, i - 1) > 0) Then
'                Cells(j, i) = Cells(j, i) + Cells(j, i - 1)
'            End If
'        Next j

            Cells(numRowsInAuthorBand + 4, i) = "=Sum(" & col(i) & "2:" & col(i) &
numRowsInAuthorBand & ")"
            'add count, avg, stdev...
            Cells(numRowsInAuthorBand + 1, i).Formula = "=Countif(" & col(i) & "2:" & col(i) &
numRowsInAuthorBand & ", "">0"")"
            If Cells(numRowsInAuthorBand + 1, i) > 0 Then
               Cells(numRowsInAuthorBand + 2, i).Formula = "=AVERAGE(" & col(i) & "2:" & col(i)
& numRowsInAuthorBand & ")"
               If Cells(numRowsInAuthorBand + 1, i) > 1 Then 'more than one so comput std deviation
                  Cells(numRowsInAuthorBand + 3, i).Formula = "=STDEV(" & col(i) & "2:" & col(i) &
numRowsInAuthorBand & ")"
               End If
            End If
         Next i

         ActiveSheet.Name = "Aff_Author_Cum_Dist_" & band & ""
         Call formatSheetForPrint

End Sub 'band authors
'--------------------------------------------------------------------------
'  Sub: CalcCumulative
'  Author: Matt Behnke
'  Created: 11/15/01
'  Description: processes the input sheet (a matrix) to calculate the cumulative number of
'          instances per time step.
'  inputs: sheetName
'
'  Outputs:
'--------------------------------------------------------------------------
Sub CalcCumulative(ByVal sheetName As String)

    numRows = CountRows(sheetName, 1)
    numCols = CountCols(sheetName, 1)

    Sheets(sheetName).Select

    If sheetName = affiliationDescMatrix Then
       For i = 2 To numRows
          cellSum = 0
          prevSum = 0
          curSum = 0
          For j = 6 To numCols
```

336

```
            prevSum = Cells(i, j - 1)
            curSum = Cells(i, j)
            cellSum = prevSum + curSum
            If cellSum > 0 Then
               Cells(i, j) = cellSum
            End If
          Next j
        Next i
'
' --- For the authors matrix zeros must be put in when
'     there is no publication in an instance
'
ElseIf sheetName = "Affiliation_authors" Or sheetName = dataSheet Then
    For i = 2 To numRows
       cellSum = 0
       prevSum = 0
       curSum = 0
       For j = 4 To numCols
          If j > 4 Then 'when not in first column
             prevSum = Cells(i, j - 1)
             curSum = Cells(i, j)
             cellSum = prevSum + curSum
             Cells(i, j) = cellSum
          Else 'in first column
             If Not Cells(i, j) > 0 Then
                Cells(i, j) = 0
             End If
          End If

       Next j
     Next i
  Else
     For i = 2 To numRows
       cellSum = 0
       prevSum = 0
       curSum = 0
       For j = 5 To numCols
          prevSum = Cells(i, j - 1)
          curSum = Cells(i, j)
          cellSum = prevSum + curSum
          If cellSum > 0 Then
             Cells(i, j) = cellSum
          End If
        Next j
      Next i
    End If

       If sheetName = dataSheet Or sheetName = "Affiliation_authors" Then 'put count, mean, and
stdev in each column
          For i = 4 To numCols 'put in the mean and std deviation for each time step
```

```
               Cells(numRows + 4, i).Formula = "=Sum(" & col(i) & "2:" & col(i) & numRows & ")"
'sum
               Cells(numRows + 1, i).Formula = "=Countif(" & col(i) & "2:" & col(i) & numRows & ",
"">0"")"

               If Cells(numRows + 1, i) > 0 Then
                  Cells(numRows + 2, i).Formula = "=AVERAGE(" & col(i) & "2:" & col(i) & numRows
& ")"

                  If Cells(numRows + 1, i) > 1 Then 'more than one so comput std deviation
                     Cells(numRows + 3, i).Formula = "=STDEV(" & col(i) & "2:" & col(i) & numRows
& ")"

                  End If
               End If
            Next i
         Else
            'put in the sum of the columns
            For i = 4 To numCols
               Cells(numRows + 1, i).Formula = "=Sum(" & col(i) & "2:" & col(i) & numRows & ")"
            Next i
         End If 'sheetname = datasheet
      End Sub


         '-------------------------------------------------------------------------
         '  Sub: FillBandTerms
         '  Author: Matt Behnke
         '  Created: 11/7/01
         '  Description: fills in the term instances for a band
         '  inputs: band name
         '
         '  Outputs:
         '-------------------------------------------------------------------------
      Sub FillBandTerms(ByVal band As String)

         Sheets.Add After:=Worksheets(Worksheets.Count)

         counter = 2
         affiliationDescMatrix = "descriptor_matrix_affil"
         numRowsInBand = CountRows("Affiliation_Cum_Dist_" & band, 1)
         numColumnsInTerms = CountCols(affiliationDescMatrix, 1)

         Sheets(Worksheets.Count).Select
         currSheetName = ActiveSheet.Name
         Columns("C:C").ColumnWidth = 32.43

         Sheets(currSheetName).Move Before:=Sheets("" & band & "_Stats")

         'header
         Cells(1, 1) = Sheets(affiliationDescMatrix).Cells(1, 1)
         Cells(1, 2) = Sheets(affiliationDescMatrix).Cells(1, 2)
         Cells(1, 3) = Sheets(affiliationDescMatrix).Cells(1, 3)
```

338

```
        For i = 4 To numColumnsInTerms - 1 'copy time interval header
            Cells(1, i) = Sheets("descriptor_matrix_affil").Cells(1, i + 1)
        Next i

        'fill in the terms and instances...
        For i = 2 To numRowsInBand 'copy rows from datasheet into band
            affiliationName = Sheets("Affiliation_Cum_Dist_" & band).Cells(i, 3).Value
            For j = 2 To CountRows(affiliationDescMatrix, 1)
                If Sheets(affiliationDescMatrix).Cells(j, 4) = affiliationName Then

                    rowInAffiliationDescMatrix = j
                    termName = Sheets(affiliationDescMatrix).Cells(rowInAffiliationDescMatrix, 3)

                    'check to see if term exists already on band's list of terms
                    termRowInBand = findStringRowInSheet(currSheetName, termName, 3)

                    If termRowInBand > 0 Then

                        Cells(termRowInBand,    2)    =    Cells(termRowInBand,    2)    +
Sheets(affiliationDescMatrix).Cells(rowInAffiliationDescMatrix, 2)
                        cellSum = 0
                        prevSum = 0
                        curSum = 0

                        If   Sheets(affiliationDescMatrix).Cells(rowInAffiliationDescMatrix,   1).Value   <
Cells(termRowInBand, 1) Then
                            Cells(termRowInBand,                       1)                       =
Sheets(affiliationDescMatrix).Cells(rowInAffiliationDescMatrix, 1).Value
                        End If

                        For z = 4 To numColumnsInTerms 'add the values for each time time to what aleady
exists
                            If z > 4 Then 'add cumulative sum of term instances (previous + current +
numInstances)
                                'prevSum = Cells(termRowInBand, z - 1)
                                If  Sheets(affiliationDescMatrix).Cells(rowInAffiliationDescMatrix, z + 1) > 0
Then
                                    curSum       =       Cells(termRowInBand,       z)       +
Sheets(affiliationDescMatrix).Cells(rowInAffiliationDescMatrix, z + 1)

                                Else
                                    curSum = 0
                                End If
                                'cellSum = prevSum + curSum
                                If curSum > 0 Then
                                    Cells(termRowInBand, z) = curSum
                                End If
                            Else
```

339

```
                              If  Sheets(affiliationDescMatrix).Cells(rowInAffiliationDescMatrix,  z  +  1)  >  0
Then
                                   Cells(termRowInBand,      z)      =      Cells(termRowInBand,      z)      +
Sheets(affiliationDescMatrix).Cells(rowInAffiliationDescMatrix, z + 1)
                              End If
                           End If
                         Next z
                    Else 'term not found
                      Cells(counter,  1)  =  Sheets(affiliationDescMatrix).Cells(rowInAffiliationDescMatrix,
1)
                              Cells(counter,  2)  =  Sheets(affiliationDescMatrix).Cells(rowInAffiliationDescMatrix,
2)
                              Cells(counter,  3)  =  Sheets(affiliationDescMatrix).Cells(rowInAffiliationDescMatrix,
3)
                      For z = 4 To numColumnsInTerms
                         If  z  >  4  Then  'add  cumulative  sum  of  term  instances  (previous  +  current  +
numInstances)

                              'prevSum = Cells(counter, z - 1)
                              curSum            =            Cells(counter,            z)            +
Sheets(affiliationDescMatrix).Cells(rowInAffiliationDescMatrix, z + 1)
                              'cellSum = prevSum + curSum

                              If curSum > 0 Then
                                 Cells(counter, z) = curSum
                              End If
                              Else
                                 If  Sheets(affiliationDescMatrix).Cells(rowInAffiliationDescMatrix,  z  +  1)  >  0
Then
                                    Cells(counter,       z)       =       Cells(counter,       z)       +
Sheets(affiliationDescMatrix).Cells(rowInAffiliationDescMatrix, z + 1)
                                 End If 'elimates zeros
                              End If 'z = 4
                           Next z
                      counter = counter + 1
                      End If 'if-found-else-not
                    End If ' affiliation name matches
                Next j
             Next i

             numRows = CountRows(ActiveSheet.Name, 1)
             numCols = CountCols(ActiveSheet.Name, 1)
               For i = 4 To numCols
                 Cells(numRows + 1, i).Formula = "=Sum(" & col(i) & "2:" & col(i) & numRows & ")"
               Next i
             'Call CalcCumulative(ActiveSheet.Name)

             ActiveSheet.Name = "Term_Dist_" & band & ""
             Call formatSheetForPrint
```

```
End Sub 'fill band terms
'----------------------------------------------------------------------
'  Sub: FillBandTermsEntropy
'  Author: Matt Behnke
'  Created: 11/17/01
'  Description: computes the entropy of a band's terms.
'              and the contribution of the band..
'  inputs: band name
'
'  Outputs:
'----------------------------------------------------------------------
Sub FillBandTermsEntropy(ByVal band As String)

    Sheets.Add After:=Worksheets(Worksheets.Count)


    numRows = CountRows("Term_Dist_" & band & "", 1)
    numColumns = CountCols("Term_Dist_" & band & "", 1)

    Sheets(Worksheets.Count).Select
    currSheetName = ActiveSheet.Name
    Columns("C:C").ColumnWidth = 32.43

    Sheets(currSheetName).Move Before:=Sheets("" & band & "_Stats")

    numRowsWorld = CountRows(descriptorMatrixSheet, 1)

    'copy term distribution sheet for entropy
    Worksheets("Term_Dist_" & band & "").Range("A1:" & col(numColumns) & numRows).Copy
Destination:=Worksheets(currSheetName).Range("A1")


    For i = 2 To numRows
        termName = Sheets(currSheetName).Cells(i, 3)
        termCount = Sheets(currSheetName).Cells(i, 2)
        termRowInWorldEntropy = findStringRowInSheet(worldEntropySheet, termName, 3)
        termRowInDescriptorMatrix = termRowInWorldEntropy

        For z = 4 To numColumns
            If Sheets(currSheetName).Cells(i, z).Value >= 1 Then
                termCountInBandInStep = Sheets(currSheetName).Cells(i, z)

                sumInstancesBand = Sheets("Term_Dist_" & band & "").Cells(numRows + 1, z)
                pTerm = termCountInBandInStep / sumInstancesBand
                entropyTerm = -pTerm * (Log(pTerm) / Log(2))

                Sheets(currSheetName).Cells(i, z) = entropyTerm

            End If
        Next z
```

```
      Next i

   Sheets(currSheetName).Select
      Cells(numRows + 1, 3) = "Sum"
      Cells(numRows + 2, 3) = "Contribution"
      Cells(numRows + 3, 3) = "Difference"

      For i = 4 To numColumns
         Cells(numRows + 1, i).Formula = "=Sum(" & col(i) & "2:" & col(i) & numRows & ")"

         numInstancesWorld = Sheets(descriptorMatrixSheet).Cells(numRowsWorld + 1, i)
         numInstancesBand = Sheets("Term_Dist_" & band & "").Cells(numRows + 1, i)
         If numInstancesBand > 0 Then
            ratio1 = numInstancesWorld / numInstancesBand
            ratio2 = numInstancesBand / numInstancesWorld
            entropySum = Cells(numRows + 1, i)
            contributionOfBand = ratio2 * entropySum + (ratio2 * (Log(ratio1) / Log(2)))
            Cells(numRows + 2, i) = contributionOfBand
            Cells(numRows + 3, i) = Abs(entropySum - contributionOfBand)
         Else
            Cells(numRows + 2, i) = 0
            Cells(numRows + 3, i) = 0
         End If
      Next i

   ActiveSheet.Name = "Term_Entropy_Dist_" & band & ""
   Call formatSheetForPrint

End Sub 'fill band terms entropy
'-------------------------------------------------------------------------
' Sub: affiliationBandSummary
' Author: Matt Behnke
' Created: 11/30/01
' Description: creates the summary sheet for the band..
'          shows step, num of recors, authors, terms, entropy..
' inputs: band - the name of the band
' Outputs: none
'-------------------------------------------------------------------------
Sub affiliationBandSummary(ByVal band As String)

   numColumns = CountCols("Term_Entropy_Dist_" & band, 1)
   numRowsAffiliation = CountRows("Affiliation_Cum_Dist_" & band, 1)
   numRowsAuthor = CountRows("Aff_Author_Cum_Dist_" & band, 1)
   numRowsTermDist = CountRows("Term_Dist_" & band, 1)
   numRowsTermEntropy = CountRows("Term_Entropy_Dist_" & band, 1)

   Sheets.Add After:=Worksheets(Worksheets.Count)
   Sheets(Worksheets.Count).Select
   currSheetName = ActiveSheet.Name
```

```
Sheets(currSheetName).Move Before:=Sheets(band & "_Stats")

Sheets(currSheetName).Select
ActiveSheet.StandardWidth = 13

Cells(1, 1) = " "
Cells(2, 1) = " "

Cells(2, 3) = "Instances (Previous + Current)"

Cells(3, 1) = "Step"
Cells(3, 2) = "interval"

Cells(3, 3) = "Records"
Cells(3, 4) = "Authors"
Cells(3, 5) = "Terms"

Cells(3, 6) = "Entropy"
Cells(3, 7) = "Contribution"
Cells(3, 8) = "Difference"
Cells(3, 9) = "Rec / Author"

For i = 4 To numColumns
    Cells(i, 1) = i - 3
    Cells(i, 2) = Sheets("Term_Dist_" & band).Cells(1, i)

    Cells(i, 3).Value = "=SUM(Affiliation_Cum_Dist_" & band & "!" & col(i) & "$2:" & col(i)
& "$" & numRowsAffiliation & ")"

    Cells(i, 4).Value = "=SUM(Aff_Author_Cum_Dist_" & band & "!" & col(i) & "$2:" & col(i)
& "$" & numRowsAuthor & ")"
    Cells(i, 5).Value = "=SUM(Term_Dist_" & band & "!" & col(i) & "$2:" & col(i) & "$" &
numRowsTermDist & ")"

    Cells(i, 6) = Sheets("Term_Entropy_Dist_" & band).Cells(numRowsTermEntropy + 1, i)
    Cells(i, 7) = Sheets("Term_Entropy_Dist_" & band).Cells(numRowsTermEntropy + 2, i)
    Cells(i, 8) = Sheets("Term_Entropy_Dist_" & band).Cells(numRowsTermEntropy + 3, i)
    If Cells(i, 4) > 0 Then
        Cells(i, 9) = Cells(i, 3) / Cells(i, 4)
    End If
Next i

ActiveSheet.Name = "Affiliation_Summary_" & band

End Sub

'-------------------------------------------------------------------------
' Sub: affiliationSummary
' Author: Matt Behnke
' Created: 11/30/01
```

343

```
' added stuff: 2/1/02
' Description: creates the world affilation summary sheet..
'            this is the first part.. the second part puts in the temp poly and the pressure equations
'            after fill months has been run on the sheet.........
' inputs: none
' Outputs: none
'-------------------------------------------------------------------------
Sub affiliationSummary()

    numColumns = CountCols(worldEntropySheet, 1)
    numRowsAffiliation = CountRows(dataSheet, 1)
    numRowsAuthor = CountRows("Affiliation_authors", 1)
    numRowsTermDist = CountRows(descriptorMatrixSheet, 1)
    numRowsTermEntropy = CountRows(worldEntropySheet, 1)

    Sheets.Add After:=Worksheets(Worksheets.Count)
    Sheets(Worksheets.Count).Select
    currSheetName = ActiveSheet.Name

    Sheets(currSheetName).Move After:=Sheets(Sheets.Count)

    Sheets(currSheetName).Select
    ActiveSheet.StandardWidth = 13


    Cells(1, 1) = " "
    Cells(2, 1) = " "

    Cells(2, 3) = "Instances (Previous + Current)"

    Cells(3, 1) = "Step"
    Cells(3, 2) = "interval"

    Cells(3, 3) = "Records"
    Cells(3, 4) = "Authors (v_X)"
    Cells(3, 5) = "Rec / Author"
    Cells(3, 6) = "Terms X"
    Cells(3, 7) = "Terms Y"
    Cells(3, 8) = "S(X)"
    Cells(3, 9) = "S(Y)"
    Cells(3, 10) = "S(X,Y)"
    Cells(3, 11) = "S(X;Y)"
    Cells(3, 12) = "delta_n_x"
    Cells(3, 13) = "delta_s_x"
    Cells(3, 14) = "T_X Saboe Degrees"
    Cells(3, 15) = "delta_n_y"
    Cells(3, 16) = "delta_s_y"
    Cells(3, 17) = "v_Y_nodes"
    Cells(3, 18) = "pressure_n per node"
```

```
        For i = 4 To numColumns
           Cells(i, 1) = i - 3
           Cells(i, 2) = Sheets(dataSheet).Cells(1, i)

           If i > 4 Then
              Cells(i, 3).Value = "=SUM(" & dataSheet & "!" & col(i) & "$2:" & col(i) & "$" &
numRowsAffiliation & ")" '+ C" & i - 1
              Else
                 Cells(i, 3).Value = "=SUM(" & dataSheet & "!" & col(i) & "$2:" & col(i) & "$" &
numRowsAffiliation & ")"
              End If

           If i > 4 Then
              Cells(i, 4).Value = "=SUM(Affiliation_authors!" & col(i) & "$2:" & col(i) & "$" &
numRowsAuthor & ")" ' + D" & i - 1
              Else
                 Cells(i, 4).Value = "=SUM(Affiliation_authors!" & col(i) & "$2:" & col(i) & "$" &
numRowsAuthor & ")"
              End If

           Cells(i, 5) = Cells(i, 3) / Cells(i, 4)
           Cells(i, 6).Value = "=SUM(" & descriptorMatrixSheet & "!" & col(i) & "$2:" & col(i) & "$"
& numRowsTermDist & ")"
           Cells(i, 7).Value = "=SUM(" & descriptorMatrixSheetY & "!" & col(i) & "$2:" & col(i) &
"$" & numRowsTermDist & ")"
           Cells(i, 8) = Sheets(worldEntropySheet).Cells(numRowsTermEntropy + 1, i)
           Cells(i, 9) = Sheets(worldEntropySheetY).Cells(numRowsTermEntropy + 1, i)
           Cells(i, 10) = Sheets(worldEntropySheet).Cells(numRowsTermEntropy + 1, numColumns)
           Cells(i, 11) = "=" & col(8) & i & "+" & col(9) & i & "-" & col(10) & i 'Cells(i, 8) + Cells(i,
9) - Cells(i, 10)

           If i > 4 Then
              Cells(i, 12) = "=" & col(6) & i & "-" & col(6) & i - 1 'cells(i, 6) - cells (i-1,6) delta_n_y
              Cells(i, 13) = "=" & col(9) & i - 1 & "-" & col(9) & i 'cells(i, 9) - cells (i-1,9) delta_s_x
              Cells(i, 14) = "=" & col(12) & i & "/" & col(13) & i 'cells(i, 12) / cells(i, 13) T_X
              Cells(i, 15) = "=" & col(7) & i & "-" & col(7) & i - 1 'cells(i, 7) - cells (i-1,7) delta_n_y
              Cells(i, 16) = "=" & col(8) & i & "-" & col(8) & i - 1 'cells(i, 8) - cells(i-1, 8) delta_s_y
           End If

           Cells(i, 17) = Sheets("Affiliation_authors").Cells(numRowsAuthor + 1, numColumns) -
Cells(i, 4)
           Cells(i, 18) = "=" & col(6) & i & "/" & col(4) & i 'cells(i,6) / cells(i,4) terms X / author X
        Next i

        Cells(4, 3).Select 'freeze panes
        ActiveWindow.FreezePanes = True

        ActiveSheet.Name = "Affiliation_Summary"

        Call fillMonthsRow("Affiliation_Summary", 4)
```

345

```
        Call fillMonthsRow("Affiliation_Summary", 4)
        'Call CopyInteractingSystemsGraphs(ActiveSheet.Name, numColumns)

    End Sub

    '-------------------------------------------------------------------------
    '  Sub: affiliationSummaryPart2
    '  Author: Matt Behnke
    '  Created: 2/1/02
    '  Description: after fillmonths has been ran this procedure copies the appropriate graphs
    '               interactive systems graphs and temp / pressure graphs
    '               uses the trendline equations from the system graph to calculate
    '               temp_polynomial and
    '               the pressure equation
    '  inputs: none
    '  Outputs: none
    '-------------------------------------------------------------------------
    Sub affiliationSummaryPart2()

        source = "Affiliation_Summary"
        numRows = CountRows(source, 1)

        Sheets(source).Cells(3, 19) = "S(X) calculated"
        Sheets(source).Cells(3, 20) = "S(Y) calculated"
        Sheets(source).Cells(3, 21) = "delta S(X) calculated"
        Sheets(source).Cells(3, 22) = "n(X) calculated"
        Sheets(source).Cells(3, 23) = "delta_n_x_calculated"
        Sheets(source).Cells(3, 24) = "T_X Saboe Deg. Polynomial"

        Call CopyInteractingSystemsGraphs("World")

        trendlineA = Sheets(source).Cells(1, 19)
        sx_a = firstPartTrendEq(trendlineA)
        sx_b = secondPartTrendEq(trendlineA)
        'sx_a = firstPartPolyTrendEq(trendlineA)
        'sx_b = secondPartPolyTrendEq(trendlineA)
        'sx_c = thirdPartPolyTrendEq(trendlineA)

        trendlineB = Sheets(source).Cells(1, 20)
        sy_a = firstPartPolyTrendEq(trendlineB)
        sy_b = secondPartPolyTrendEq(trendlineB)
        sy_c = thirdPartPolyTrendEq(trendlineB)

        trendline_nX = Sheets(source).Cells(1, 22)
        nx_a = firstPartTrendEq(trendline_nX)
        nx_b = secondPartTrendEq(trendline_nX)

        For i = 4 To numRows
            k = Sheets(source).Cells(i, 1)
            'sX & sY calculated
```

```vba
            Sheets(source).Cells(i, 19) = sx_a * k ^ sx_b 'power equation of entropy
            'Sheets(source).Cells(i, 19) = sx_a * k ^ 2 + sx_b * k + sx_c
            Sheets(source).Cells(i, 20) = sy_a * k ^ 2 + sy_b * k + sy_c
            'nX calculated
            Sheets(source).Cells(i, 22) = nx_a * (k ^ nx_b)


            If Sheets(source).Cells(i - 1, 6).Font.ColorIndex = 3 Then
                'find the first row of the same value
                x = i - 1
                While Sheets(source).Cells(x, 6).Font.ColorIndex = 3
                    x = x - 1
                Wend
                previous_SY = Sheets(source).Cells(x, 9) 'S(Y) from previous step
                previous_nX = Sheets(source).Cells(x, 6) 'number of terms in previous step
            Else
                previous_SY = Sheets(source).Cells(i - 1, 9) 'S(Y) from previous step
                previous_nX = Sheets(source).Cells(i - 1, 6) 'number of terms in previous step
            End If

            If i > 4 Then
                'check to see if current S(Y) or current n(X) (num terms) is the same as previous
                'if so then place the value of the calculated S(Y) or n(X) into that spot of similarity
                'mark the spot in red where a calculated value has been substituted.
                If Sheets(source).Cells(i, 9) = previous_SY Then
                    Sheets(source).Cells(i, 9) = "=" & col(20) & i 'equals calc'ed value of S(Y)
                    Sheets(source).Cells(i, 9).Font.ColorIndex = 3
                End If
                If Sheets(source).Cells(i, 6) = previous_nX Then
                    Sheets(source).Cells(i, 6) = "=" & col(22) & i 'equals calc'ed value of n(X)
                    Sheets(source).Cells(i, 6).Font.ColorIndex = 3
                End If

                'delta S(X)_calculated
                Sheets(source).Cells(i, 21) = "=" & col(19) & i & "-" & col(19) & i - 1 'cells(i,19) - cells(i-
1,19)

                'delta n(X)_calculated
                Sheets(source).Cells(i, 23) = "=" & col(22) & i & "-" & col(22) & i - 1 'cells(i,22) - cells(i-
1,22)

                't(x)_poly = n(X)/S(X)
                Sheets(source).Cells(i, 24) = "=" & col(23) & i & "/" & col(21) & i 'cells(i,23) / cells(i,21)
            End If
        Next i

End Sub

'-------------------------------------------------------------------------
'  Sub: affiliationSummaryPart3
'  Author: Matt Behnke
'  Created: 2/4/02
```

```
'   Description: copies the temp / pressure graphs uses trendline equations of temp_poly and
pressure to get the
'            the pressure equation
' inputs: none
' Outputs: none
'------------------------------------------------------------------------
Sub affiliationSummaryPart3()

   source = "Affiliation_Summary"
   numRows = CountRows(source, 1)

   Sheets(source).Cells(3, 25) = "Press f(T)"
   Sheets(source).Cells(1, 24) = "m_P"
   Sheets(source).Cells(2, 24) = "b_P"
   Sheets(source).Cells(1, 26) = "m_T"
   Sheets(source).Cells(2, 26) = "b_T"

   Call CopyTempPressGraphs("World")

   trendline_Tpoly = Sheets(source).Cells(1, 27)
   m_t = firstPartTrendEq(trendline_Tpoly)
   b_t = secondPartLinearTrendEq(trendline_Tpoly)
   Sheets(source).Cells(1, 27) = m_t
   Sheets(source).Cells(2, 27) = b_t

   trendline_Press = Sheets(source).Cells(1, 25)
   m_p = firstPartTrendEq(trendline_Press)
   b_p = secondPartLinearTrendEq(trendline_Press)
   Sheets(source).Cells(1, 25) = m_p
   Sheets(source).Cells(2, 25) = b_p


   For i = 5 To numRows
      Tx_poly = Sheets(source).Cells(i, 24)
      Sheets(source).Cells(i, 25) = b_p + (m_p / m_t) * (Tx_poly - b_t)
   Next i

'copy third Graph World_Press_vs_Temp_Saboe
   Application.DisplayAlerts = False

   Windows("AffiliationMacro.xls").Activate
   Sheets("World_Press_vs_Temp_Saboe").Select
   Sheets("World_Press_vs_Temp_Saboe").Copy
After:=Workbooks(currFilename).Sheets(source)
   ActiveChart.SeriesCollection(1).Select

   'Pressure per node
   ActiveChart.SeriesCollection(1).Values = "=" & source & "!R" & 5 & "C25:R" & numRows &
"C25"
```

```
          ActiveChart.SeriesCollection(1).XValues = "=" & source & "!R" & 5 & "C24:R" & numRows
& "C24"
          'T(x) poly:
          ActiveChart.SeriesCollection(2).Values = "=" & source & "!R" & 5 & "C18:R" & numRows &
"C18"

          ActiveChart.SeriesCollection(2).XValues = "=" & source & "!R" & 5 & "C24:R" & numRows
& "C24"


          Application.DisplayAlerts = True

     End Sub


     '------------------------------------------------------------------------
     '  Sub: copyTempPressGraphs
     '  Author: Matt Behnke
     '  Created: 2/4/02
     '  Description: copies the interacting systems graphs from the affiliation macro workbook.
     '  inputs: band name
     '
     '  Outputs:
     '------------------------------------------------------------------------
     Sub CopyTempPressGraphs(ByVal band As String)

        Application.DisplayAlerts = False

        If band = "World" Then
           source = "Affiliation_Summary"
        Else
           source = "Affiliation_Summary_" & band
        End If

        numRows = CountRows(source, 1)

     'GRAPH ONE XY_Temp

        Windows("AffiliationMacro.xls").Activate
        Sheets("XY_Temp").Select
        Sheets("XY_Temp").Copy After:=Workbooks(currFilename).Sheets(source)
        ActiveChart.SeriesCollection(1).Select

        j = 4
        While Not (i > 0#)   'if the first time step's mean is zero find the step that doesnt have 0
           i = Sheets(source).Cells(j, 3).Value
           If Not (i > 0) Then
              j = j + 1
           End If
        Wend

        'X-Category
```

```
            'msgs per node
            ActiveChart.SeriesCollection(1).Values = "=" & source & "!R" & j & "C18:R" & numRows &
"C18"

            ActiveChart.SeriesCollection(1).XValues = "=" & source & "!R" & j & "C1:R" & numRows &
"C1"

            't(x)
            ActiveChart.SeriesCollection(2).Values = "=" & source & "!R" & j & "C14:R" & numRows &
"C14"

            ActiveChart.SeriesCollection(2).XValues = "=" & source & "!R" & j & "C1:R" & numRows &
"C1"

            't(x) poly
            ActiveChart.SeriesCollection(3).Values = "=" & source & "!R" & j & "C24:R" & numRows &
"C24"

            ActiveChart.SeriesCollection(3).XValues = "=" & source & "!R" & j & "C18:R" & numRows
& "C18"


            With ActiveChart.SeriesCollection(2).Trendlines(1)
            'put trendline equation onto stats sheet for T(X)_poly
            .DisplayEquation = True
            .DisplayRSquared = True

            End With

            With ActiveChart.SeriesCollection(3).Trendlines(1)
            'put trendline equation onto stats sheet for T(X)_poly
            .DisplayEquation = True
            .DisplayRSquared = True
             Worksheets(source).Cells(1, 27).Value = .DataLabel.Text
            End With

           'copy second graph XY_Press
            Windows("AffiliationMacro.xls").Activate
            Sheets("XY_Press").Select
            Sheets("XY_Press").Copy After:=Workbooks(currFilename).Sheets(source)

            ActiveChart.SeriesCollection(1).Select

            'Pressure per node
            ActiveChart.SeriesCollection(1).Values = "=" & source & "!R" & j & "C18:R" & numRows &
"C18"

            ActiveChart.SeriesCollection(1).XValues = "=" & source & "!R" & j & "C1:R" & numRows &
"C1"

            'T(x) poly:
            ActiveChart.SeriesCollection(2).Values = "=" & source & "!R" & j & "C24:R" & numRows &
"C24"

            ActiveChart.SeriesCollection(2).XValues = "=" & source & "!R" & j & "C18:R" & numRows
& "C18"


            With ActiveChart.SeriesCollection(2).Trendlines(1)
            'put trendline equation onto stats sheet for pressure fit
```

```
      .DisplayEquation = True
      .DisplayRSquared = True
      End With

      With ActiveChart.SeriesCollection(1).Trendlines(1)
      'put trendline equation onto stats sheet for pressure fit
      .DisplayEquation = True
      .DisplayRSquared = True
       Worksheets(source).Cells(1, 25).Value = .DataLabel.Text
      End With

      Application.DisplayAlerts = True

End Sub 'copy temp/press graphs



'--------------------------------------------------------------------------
'  Sub: copyInteractingSystemsGraphs
'  Author: Matt Behnke
'  Created: 2/1/02
'  Description: copies the interacting systems graphs from the affiliation macro workbook.
'  inputs: band name
'
'  Outputs:
'--------------------------------------------------------------------------
Sub CopyInteractingSystemsGraphs(ByVal band As String)

    Application.DisplayAlerts = False

   If band = "World" Then
      source = "Affiliation_Summary"
   Else
      source = "Affiliation_Summary_" & band
   End If

   numRows = CountRows(source, 1)

'GRAPH ONE S_2Interacting Systems

    Windows("AffiliationMacro.xls").Activate
    Sheets("S_2Interacting systems").Select
    Sheets("S_2Interacting systems").Copy After:=Workbooks(currFilename).Sheets(source)
    ActiveChart.SeriesCollection(1).Select

   j = 4
   While Not (i > 0#)   'if the first time step's mean is zero find the step that doesnt have 0
      i = Sheets(source).Cells(j, 3).Value
      If Not (i > 0) Then
         j = j + 1
      End If
```

```
        Wend

        'X-Category

        'S(Y)
        ActiveChart.SeriesCollection(1).Values = "=" & source & "!R" & j & "C9:R" & numRows &
"C9"

        ActiveChart.SeriesCollection(1).XValues = "=" & source & "!R" & j & "C6:R" & numRows &
"C6"

        'S(X):
        ActiveChart.SeriesCollection(2).Values = "=" & source & "!R" & j & "C8:R" & numRows &
"C8"

        ActiveChart.SeriesCollection(2).XValues = "=" & source & "!R" & j & "C6:R" & numRows &
"C6"

        'S(X,Y)
        ActiveChart.SeriesCollection(3).Values = "=" & source & "!R" & j & "C10:R" & numRows &
"C10"

        ActiveChart.SeriesCollection(3).XValues = "=" & source & "!R" & j & "C6:R" & numRows &
"C6"

        'S(X;Y)
        ActiveChart.SeriesCollection(4).Values = "=" & source & "!R" & j & "C11:R" & numRows &
"C11"

        ActiveChart.SeriesCollection(4).XValues = "=" & source & "!R" & j & "C6:R" & numRows &
"C6"


        With ActiveChart.SeriesCollection(1).Trendlines(1)
        'put trendline equation onto stats sheet for S(y)
        .DisplayEquation = True
        .DisplayRSquared = True
         Worksheets(source).Cells(1, 20).Value = .DataLabel.Text
        End With

        With ActiveChart.SeriesCollection(2).Trendlines(1)
        'put trendline equation onto stats sheet for S(x)
        .DisplayEquation = True
        .DisplayRSquared = True
         Worksheets(source).Cells(1, 19).Value = .DataLabel.Text
        End With

       'copy second graph World_(X)Temp_S_2
        Windows("AffiliationMacro.xls").Activate
        Sheets("World_(X)Temp_S_2").Select
        Sheets("World_(X)Temp_S_2").Copy After:=Workbooks(currFilename).Sheets(source)

        ActiveChart.SeriesCollection(1).Select

        'S(X) vs T_X
        ActiveChart.SeriesCollection(1).Values = "=" & source & "!R" & j & "C8:R" & numRows &
"C8"
```

```vb
            ActiveChart.SeriesCollection(1).XValues = "=" & source & "!R" & j & "C14:R" & numRows
& "C14"
            'S(X;Y):
            ActiveChart.SeriesCollection(2).Values = "=" & source & "!R" & j & "C11:R" & numRows &
"C11"
            ActiveChart.SeriesCollection(2).XValues = "=" & source & "!R" & j & "C6:R" & numRows &
"C6"

            With ActiveChart.SeriesCollection(1).Trendlines(1)
            'put trendline equation onto stats sheet for S(y)
            .DisplayEquation = True
            .DisplayRSquared = True

            End With

            With ActiveChart.SeriesCollection(2).Trendlines(1)
            'put trendline equation onto stats sheet for S(x)
            .DisplayEquation = True
            .DisplayRSquared = True

            End With

        'copy third Graph n_Msg_2Interacting systems
            Windows("AffiliationMacro.xls").Activate
            Sheets("n_Msg_2Interacting systems").Select
            Sheets("n_Msg_2Interacting systems").Copy After:=Workbooks(currFilename).Sheets(source)
            ActiveChart.SeriesCollection(1).Select

            'n_X
            ActiveChart.SeriesCollection(1).Values = "=" & source & "!R" & j & "C6:R" & numRows &
"C6"

            With ActiveChart.SeriesCollection(1).Trendlines(1)
            'put trendline equation onto stats sheet for S(x)
            .DisplayEquation = True
            .DisplayRSquared = True
             Worksheets(source).Cells(1, 22).Value = .DataLabel.Text
            End With

            'n_Y
            ActiveChart.SeriesCollection(2).Values  =  "=Affiliation_Summary!R"  &  j  &  "C7:R"  &
numRows & "C7"

            Application.DisplayAlerts = True

        End Sub 'copy interacting systems graphs

            '-------------------------------------------------------------------------
            ' Sub: entropySummary
            ' Author: Matt Behnke
```

```vba
' Created: 11/19/01
' Description: creates the entropy summary sheet.. for the world and all the bands,
'            shows the local and contribution entropies of each band
' inputs: none
' Outputs: none
'------------------------------------------------------------------------
Sub entropySummary()

    numColumnsInTerms = CountCols("Term_Entropy_Dist_A_Band", 1)
    numRowsAband = CountRows("Term_Entropy_Dist_A_Band", 1)
    numRowsBband = CountRows("Term_Entropy_Dist_B_Band", 1)
    numRowsCband = CountRows("Term_Entropy_Dist_C_Band", 1)
    numRowsDband = CountRows("Term_Entropy_Dist_D_Band", 1)
    numRowsWorld = CountRows(worldEntropySheet, 1)

    Sheets.Add After:=Worksheets(Worksheets.Count)
    Sheets(Worksheets.Count).Select
    currSheetName = ActiveSheet.Name

    Sheets(currSheetName).Move After:=Sheets("D_Band_Stats")

    Sheets(currSheetName).Select
    ActiveSheet.StandardWidth = 13

    Cells(1, 1) = " "
    Cells(2, 1) = " "

    Cells(3, 1) = "Step"
    Cells(3, 2) = "interval"

    Cells(3, 3) = "A_Band Entropy"
    Cells(3, 4) = "A_Band Contribution"
    Cells(3, 5) = "A_Band Difference"

    Cells(3, 6) = "B_Band Entropy"
    Cells(3, 7) = "B_Band Contribution"
    Cells(3, 8) = "B_Band Difference"

    Cells(3, 9) = "C_Band Entropy"
    Cells(3, 10) = "C_Band Contribution"
    Cells(3, 11) = "C_Band Difference"

    Cells(3, 12) = "D_Band Entropy"
    Cells(3, 13) = "D_Band Contribution"
    Cells(3, 14) = "D_Band Difference"

    Cells(3, 15) = "Sum Band Entropy"
    Cells(3, 16) = "Sum Band Contribution"
    Cells(3, 17) = "World Entropy"
    Cells(3, 18) = "Diff World & Contrib"
```

```vba
    For i = 4 To numColumnsInTerms
      Cells(i, 1) = i - 3
      Cells(i, 2) = Sheets("Term_Entropy_Dist_A_Band").Cells(1, i)

      Cells(i, 3) = Sheets("Term_Entropy_Dist_A_Band").Cells(numRowsAband + 1, i)
      Cells(i, 4) = Sheets("Term_Entropy_Dist_A_Band").Cells(numRowsAband + 2, i)
      Cells(i, 5) = Sheets("Term_Entropy_Dist_A_Band").Cells(numRowsAband + 3, i)

      Cells(i, 6) = Sheets("Term_Entropy_Dist_B_Band").Cells(numRowsBband + 1, i)
      Cells(i, 7) = Sheets("Term_Entropy_Dist_B_Band").Cells(numRowsBband + 2, i)
      Cells(i, 8) = Sheets("Term_Entropy_Dist_B_Band").Cells(numRowsBband + 3, i)

      Cells(i, 9) = Sheets("Term_Entropy_Dist_C_Band").Cells(numRowsCband + 1, i)
      Cells(i, 10) = Sheets("Term_Entropy_Dist_C_Band").Cells(numRowsCband + 2, i)
      Cells(i, 11) = Sheets("Term_Entropy_Dist_C_Band").Cells(numRowsCband + 3, i)

      Cells(i, 12) = Sheets("Term_Entropy_Dist_D_Band").Cells(numRowsDband + 1, i)
      Cells(i, 13) = Sheets("Term_Entropy_Dist_D_Band").Cells(numRowsDband + 2, i)
      Cells(i, 14) = Sheets("Term_Entropy_Dist_D_Band").Cells(numRowsDband + 3, i)

      Cells(i, 15) = Cells(i, 3) + Cells(i, 6) + Cells(i, 9) + Cells(i, 12)
      Cells(i, 16) = Cells(i, 4) + Cells(i, 7) + Cells(i, 10) + Cells(i, 13)
      Cells(i, 17) = Sheets(worldEntropySheet).Cells(numRowsWorld + 1, i)
      Cells(i, 18) = Abs(Cells(i, 17) - Cells(i, 16))
    Next i

    ActiveSheet.Name = "Entropy Summary"
End Sub

'--------------------------------------------------------------------------
'  Function: findStringRowInSheet
'  Author: Matt Behnke
'  Created: 2/28/02
'  Description: determines the row of the string in the given sheet. uses find function
'  inputs: matrixSheet, termName (descriptor), column letter of term in matrixSheet
'  Outputs: the row number
'--------------------------------------------------------------------------
Function findStringInSheet(ByVal matrixSheet As String, ByVal termName As String, ByVal
column As String) As String

    With Worksheets(matrixSheet).Range(column & ":" & column)
      Set C = .Find(termName, LookIn:=xlValues)
      If Not C Is Nothing Then
        firstAddress = C.Address
        temp = Sheets(1).Cells(1, 1)
        Sheets(1).Cells(1, 1) = firstAddress
        theRow = Sheets(1).Cells(1, 1).Characters(4, 5).Text
        Sheets(1).Cells(1, 1) = temp
```

```
          findStringInSheet = theRow
        Else
          findStringInSheet = 0
        End If
    End With




    End Function 'funciton


    '------------------------------------------------------------------------
    ' Function: findStringRowInSheet ****OBSOLETE*** Slow
    ' Author: Matt Behnke
    ' Created: 11/16/01
    ' Description: determines the row of the string in the given sheet
    ' inputs: sheetname, descriptor, column of desc in datasheet
    ' Outputs: row number where the value is found
    '------------------------------------------------------------------------
    Function findStringRowInSheet(ByVal matrixSheet As String, ByVal termName As String,
ByVal columnNum As Integer) As Integer


        foundAt = 0
        numRows = CountRows(matrixSheet, 1)
        For i = 2 To numRows 'assume column header
          If Cells(i, columnNum).Value = termName Then
             foundAt = i
             found = True
             Exit For
          End If
        Next i
        If found = True Then
          findStringRowInSheet = foundAt
        Else
          findStringRowInSheet = 0
        End If

    End Function
    '------------------------------------------------------------------------
    ' Subroutine: formatSheetForPrint
    ' Author: Matt Behnke
    ' Created: 9/19/01
    ' Description: formats the sheet to fit on one page wide (legal size paper)
    '              adds header and footer to each sheet and sets orientation to landscape
    ' inputs: none
    ' Outputs: none
    '------------------------------------------------------------------------
```

```vba
Sub formatSheetForPrint()
'column heading                                    (R11.3)
   With ActiveSheet.PageSetup
      .PrintTitleRows = "$1:$1"
      .PrintTitleColumns = ""
   End With
   ActiveSheet.PageSetup.PrintArea = "$A$1:$Y$203"
   With ActiveSheet.PageSetup
      .LeftHeader = ""
      .CenterHeader = "&A in &F"                   '(R11.4)
      .RightHeader = ""
      .LeftFooter = "&D"                           '(R11.5)
      .CenterFooter = "Page &P of &N"
      .RightFooter = ""
      .LeftMargin = Application.InchesToPoints(0.75)
      .RightMargin = Application.InchesToPoints(0.75)
      .TopMargin = Application.InchesToPoints(1)
      .BottomMargin = Application.InchesToPoints(1)
      .HeaderMargin = Application.InchesToPoints(0.5)
      .FooterMargin = Application.InchesToPoints(0.5)
      .PrintHeadings = False
      .PrintGridlines = True
      .PrintComments = xlPrintNoComments
      .CenterHorizontally = False
      .CenterVertically = False
      .Orientation = xlLandscape                   '(R11.6)
      .Draft = False
      .PaperSize = xlPaperLetter                   '(R11.1)
      .FirstPageNumber = xlAutomatic
      .Order = xlDownThenOver
      .BlackAndWhite = False
      .Zoom = False
      .FitToPagesWide = 1                          '(R11.2)
      .FitToPagesTall = 99
   End With
End Sub 'format sheet for print

Sub rSquaredtest()

Call rSquaredSheet("World")
Call rSquaredSheet("A_Band")
Call rSquaredSheet("B_Band")
Call rSquaredSheet("C_Band")
Call rSquaredSheet("D_Band")
End Sub

'--------------------------------------------------------------------------------
' rSquaredSheet
' author: Matt Behnke
' created 1/3/02
```

```
' creates a new sheet that stores the R*R values of the graphs:
'     number of publications over time and cumulative entropy over time
' Uses the information stored in the affiliation Summary sheets..
'-----------------------------------------------------------------------

Sub rSquaredSheet(ByVal band As String)

    Sheets.Add After:=Worksheets(Worksheets.Count)
    Sheets(Worksheets.Count).Select

    currSheet = ActiveSheet.Name
    Sheets(currSheet).Select

    'set the source of the data
    If band = "World" Then
        source = "Affiliation_Summary"
    Else
        source = "Affiliation_Summary_" & band
    End If

    numRows = CountRows(source, 1)

    'fill in the header information of the rsquared sheet
    Call rSquaredSheetHeader(numRows - 3, currSheet, band)

    'determine startrow
    startRow = 4
    While Not (i > 0#)   'if the first time step's value is zero find the step that isn't 0
        i = Sheets(source).Cells(startRow, 3).Value
        If Not (i > 0) Then
            startRow = startRow + 1
        End If
    Wend

    counter = 6
    For i = startRow To numRows
        'get the month
        j = 1
        While found = False
            testchar = Sheets(source).Cells(i, 2).Characters(j, 1).Text
            If testchar = "/" Then
                found = True
            Else
                j = j + 1
            End If
        Wend
        'month ends at j

        'get the year
        currentMonth = Sheets(source).Cells(i, 2).Characters(1, j - 1).Text
```

358

```vbnet
        If currentMonth < 10 Then
           theYear = Sheets(source).Cells(i, 2).Characters(5, 2).Text
        Else
           theYear = Sheets(source).Cells(i, 2).Characters(6, 2).Text
        End If

        'test the year to see if different from before
        If Not theYear = previousYear Then
           startGraphRange = i
           startStep = Sheets(source).Cells(i, 1)
           Sheets(currSheet).Cells(counter, 2) = startStep
           If theYear > 50 Then
              Sheets(currSheet).Cells(counter, 1) = "19" & theYear
           Else
              Sheets(currSheet).Cells(counter, 1) = "20" & theYear
           End If

           Call rSquaredGraph(currSheet, startGraphRange, counter, source)
           counter = counter + 1
        End If

        'update previous year value
        previousYear = theYear
        found = False
      Next i

    Sheets(currSheet).Name = band & "_rSquared_Power"
End Sub 'rSquaredSheet

    '----------------------------------------------------------------
    ' rSquaredHeader
    ' Author: Matt Behnke
    ' Created 1 / 3 / 2002
    ' creates the header columns and formatting for the rsquared sheet
    '
    '----------------------------------------------------------------
    Sub rSquaredSheetHeader(ByVal numSteps As Integer, ByVal rSquaredSheetName As String,
ByVal band As String)

      Sheets(rSquaredSheetName).Select

      Range("A1").Select
      ActiveCell.FormulaR1C1 = "R-Squared Values for Ada, " & band
      Range("A3").Select
      ActiveCell.FormulaR1C1 = numSteps & " total steps, " & numSteps / 12 & " years."
      Range("A4").Select
      ActiveCell.FormulaR1C1 = "Number of Publications"

      Range("A5").Select
      ActiveCell.FormulaR1C1 = "Year"
```

```
Range("B5").Select
ActiveCell.FormulaR1C1 = "Starting Step"
Range("C5").Select
ActiveCell.FormulaR1C1 = "R_squared"
Range("D5").Select
ActiveCell.FormulaR1C1 = "Equation"

Columns("A:J").Select
Selection.ColumnWidth = 13.29

Columns("D:D").Select
Selection.ColumnWidth = 18

Columns("F:F").Select
Selection.ColumnWidth = 18

Range("C5:D5").Select
Selection.Copy

Range("E5").Select
ActiveSheet.Paste

Range("E4").Select
Application.CutCopyMode = False
ActiveCell.FormulaR1C1 = "Entropy"

Range("A5:I5").Select
Selection.Borders(xlDiagonalDown).LineStyle = xlNone
Selection.Borders(xlDiagonalUp).LineStyle = xlNone
Selection.Borders(xlEdgeLeft).LineStyle = xlNone
Selection.Borders(xlEdgeTop).LineStyle = xlNone
With Selection.Borders(xlEdgeBottom)
   .LineStyle = xlContinuous
   .Weight = xlThin
   .ColorIndex = xlAutomatic
End With
Selection.Borders(xlEdgeRight).LineStyle = xlNone
Selection.Borders(xlInsideVertical).LineStyle = xlNone

Columns("C:C").Select
With Selection.Borders(xlEdgeLeft)
   .LineStyle = xlContinuous
   .Weight = xlThin
   .ColorIndex = xlAutomatic
End With

Columns("E:E").Select
With Selection.Borders(xlEdgeLeft)
   .LineStyle = xlContinuous
   .Weight = xlThin
```

```
        .ColorIndex = xlAutomatic
      End With

      Range("A1").Select
      Selection.Font.Bold = True
   End Sub 'rsquaredHeader



      '----------------------------------
      ' Sub: rSquaredGraph
      ' author: Matt Behnke
      ' Date: 1/3/2002
      ' uses a graph to determine the rsqurared value and equation ..
      ' uses affiliation summary sheets
      '----------------------------------------------
      Sub rSquaredGraph(ByVal rSquaredSheetName As String, ByVal startGraphRange As Integer,
ByVal counter As Integer, ByVal source As String)

      'trendType = xlLinear
      trendType = xlPower

      'number of publications
        Charts.Add
        chartName = ActiveChart.Name
        ActiveChart.ChartType = xlLineMarkers
        ActiveChart.SetSourceData source:=Sheets(source).Range( _
           "C" & startGraphRange & ":C255"), PlotBy:=xlColumns

        With ActiveChart
           .HasTitle = False
           .Axes(xlCategory, xlPrimary).HasTitle = False
           .Axes(xlValue, xlPrimary).HasTitle = False
        End With


        ActiveChart.SeriesCollection(1).Select
        ActiveChart.SeriesCollection(1).Trendlines.Add(Type:=trendType, Forward:=0, _
           Backward:=0, DisplayEquation:=True, DisplayRSquared:=True).Select

      'get trendline rsq and equation for num publications
        ActiveChart.SeriesCollection(1).Select
        With ActiveChart.SeriesCollection(1).Trendlines(1)
           trendEq = .DataLabel.Text
        End With

        Sheets(source).Select
        firstPartEq = firstPartTrendEq(trendEq)
        secondPartEq = secondPartTrendEq(trendEq)
        rSquared = rSquaredTrendEq(trendEq)
```

361

```vba
        If trendType = xlPower Then
           Sheets(rSquaredSheetName).Cells(counter, 4) = "y=" & firstPartEq & "x^" & secondPartEq
        Else
           Sheets(rSquaredSheetName).Cells(counter, 4) = "y=" & firstPartEq & "x + " & secondPartEq
        End If
        Sheets(rSquaredSheetName).Cells(counter, 3) = rSquared

     'entropy
        Sheets(chartName).Select
        If Not Sheets(source).Cells(startGraphRange, 6) > 0 And trendType = xlPower Then

            Sheets(rSquaredSheetName).Cells(counter, 5) = "N/A due to zero entropy"

        Else
           ActiveChart.SetSourceData source:=Sheets(source).Range( _
              "F" & startGraphRange & ":F255"), PlotBy:=xlColumns

        'get trendline rsq and equation for entropy
           ActiveChart.SeriesCollection(1).Select
           With ActiveChart.SeriesCollection(1).Trendlines(1)
             trendEq = .DataLabel.Text
           End With

           Sheets(source).Select
           firstPartEq = firstPartTrendEq(trendEq)
           secondPartEq = secondPartTrendEq(trendEq)
           rSquared = rSquaredTrendEq(trendEq)

           If trendType = xlPower Then
              Sheets(rSquaredSheetName).Cells(counter, 6) = "y=" & firstPartEq & "x^" &
secondPartEq
           Else
              Sheets(rSquaredSheetName).Cells(counter, 6) = "y=" & firstPartEq & "x + " &
secondPartEq
           End If
           Sheets(rSquaredSheetName).Cells(counter, 5) = rSquared

        End If


     'delete chart
        Sheets(rSquaredSheetName).Select
        Application.DisplayAlerts = False
        Sheets(chartName).Delete
        Application.DisplayAlerts = True


     End Sub 'rSquaredGraph
     '------------------------------------------------------------------------
     '  Function: rSquaredTrendEq
```

```
'  Author: Matt Behnke
'  Created: 1/3/02
'  Description: extracts the rSquared value of a trendline equation
'  inputs:   trendline equation
'  Outputs: firstpart of trendline equation
'-------------------------------------------------------------------------
Function rSquaredTrendEq(ByVal trendlineEq As String) As Double

   tempStorage = Cells(1, 1)
   Cells(1, 1) = trendlineEq

   i = 1
   While found = False
      testchar = Cells(1, 1).Characters(i, 1).Text
      If testchar = "R" Then
         found = True
      Else
      i = i + 1
      End If
    Wend

   'i = location of R
   'secondpart starts at character i plus 5..
   'num of characters = location(x) - 5
   'extract 5 characters..

   rSquaredTrendEq = Cells(1, 1).Characters(i + 5, 6).Text
   Cells(1, 1) = tempStorage

End Function ' rSquaredTrendEqu



'-------------------------------------------------------------------------
'  Function: CountRows
'  Author: ? Revised by: Matt Behnke
'  Created: ?
'  Revised: 9/10/01
'  Description: Counts the rows in the suppiled worksheet and column number
'  inputs:   sheetName - name of the sheet to count the rows in
'        colNum - number of the column to count rows in
'  Outputs: number of rows as a double
'-------------------------------------------------------------------------
Function CountRows(ByVal sheetName As String, ByVal colNum As Integer) As Double
On Error Resume Next
Dim currCell As Range, rowNum As Double

 Sheets("" & sheetName).Select

 If IsNumeric(colNum) Then
 Else
```

```
    colNum = 1
  End If

  rowNum = 1
  Set currCell = ActiveSheet.Cells(rowNum, colNum)
  Do While currCell.Value <> ""
    rowNum = rowNum + 1
    Set currCell = ActiveSheet.Cells(rowNum, colNum)
  Loop
  CountRows = rowNum - 1
End Function 'CountRows
'------------------------------------------------------------------------
'  Function: CountCols
'  Author: ? Revised by: Matt Behnke
'  Created: ?
'  Revised: 9/10/01
'  Description: Counts the rows in the suppiled worksheet and column number
'  inputs:   sheetName - name of the sheet to count the columns in
'          rowNum - number of the row to count columns in
'  Outputs: number of columns as a double
'------------------------------------------------------------------------
Function CountCols(ByVal sheetName As String, ByVal rowNum As Integer) As Integer
On Error Resume Next
Dim currCell As Range, colNum As Integer

  Sheets("" & sheetName).Select

  If IsNumeric(rowNum) Then
  Else
    rowNum = 1
  End If
  colNum = 1
  Set currCell = ActiveSheet.Cells(rowNum, colNum)
  Do While currCell.Value <> ""
    colNum = colNum + 1
    Set currCell = ActiveSheet.Cells(rowNum, colNum)
  Loop
  CountCols = colNum - 1
End Function 'CountCols
'------------------------------------------------------------------------
'  Function: firstPartTrendEq
'  Author: Matt Behnke
'  Created: 11/13/01
'  Description: extracts the first part of the given POWER trendline equation, works w/ linear
'  inputs:   trendline equation
'  Outputs: firstpart of trendline equation
'------------------------------------------------------------------------
Function firstPartTrendEq(ByVal trendlineEq As String) As Double

  tempStorage = Sheets(dataSheet).Cells(1, 1)
```

```
    Sheets(dataSheet).Cells(1, 1) = trendlineEq

  i = 1
  While found = False
     testchar = Sheets(dataSheet).Cells(1, 1).Characters(i, 1).Text
     If testchar = "x" Then
        found = True
     Else
     i = i + 1
     End If
   Wend

  'i = location of x
  'firstpart = starts at character 5
  'num of characters = location(x) - 5

  firstPartTrendEq = Sheets(dataSheet).Cells(1, 1).Characters(5, i - 5).Text
  Sheets(dataSheet).Cells(1, 1) = tempStorage

End Function ' first part trendline
'---------------------------------------------------------------------------
'  Function: secondPartTrendEq
'  Author: Matt Behnke
'  Created: 11/13/01
'  Description: extracts the second part of the given POWER trendline equation
'  inputs:   trendline equation
'  Outputs: firstpart of trendline equation
'---------------------------------------------------------------------------
Function secondPartTrendEq(ByVal trendlineEq As String) As Double

  tempStorage = Sheets(dataSheet).Cells(1, 1)
  Sheets(dataSheet).Cells(1, 1) = trendlineEq

  i = 1
  While found = False
     testchar = Sheets(dataSheet).Cells(1, 1).Characters(i, 1).Text
     If testchar = "x" Then
        found = True
     Else
     i = i + 1
     End If
   Wend

  'i = location of x
  'secondpart starts at character i plus 1..
  'num of characters = location(x) - 5
  'extract 5 characters..

  secondPartTrendEq = Sheets(dataSheet).Cells(1, 1).Characters(i + 1, 5).Text
  Sheets(dataSheet).Cells(1, 1) = tempStorage
```

End Function ' secondPart eq

```
'------------------------------------------------------------------------
'  Function: secondPartLinearTrendEq
'  Author: Matt Behnke
'  Created: 2/5/02
'  Description: extracts the second part of the given linear trendline equation
'  inputs:   trendline equation
'  Outputs: secondPart of trendline equation
'------------------------------------------------------------------------
Function secondPartLinearTrendEq(ByVal trendlineEq As String) As Double

    tempStorage = Sheets(dataSheet).Cells(1, 1)
    Sheets(dataSheet).Cells(1, 1) = trendlineEq

   i = 1
   While found = False
      testchar = Sheets(dataSheet).Cells(1, 1).Characters(i, 1).Text
      If testchar = "x" Then
         found = True
      Else
      i = i + 1
      End If
    Wend

   'i = location of x
   'secondpart starts at character i plus 1..
   'num of characters = location(x) +6
   ' 4.143x + 2.4441
   '      ^^^^^^^^^
   'extract 9 characters..

    secondPartLinearTrendEq = Sheets(dataSheet).Cells(1, 1).Characters(i + 1, 9).Text
    Sheets(dataSheet).Cells(1, 1) = tempStorage

End Function ' secondPart eq


'------------------------------------------------------------------------
'  Function: firstPartPolyTrendEq
'  Author: Matt Behnke
'  Created: 2/1/02
'  Description: extracts the first part of the given trendline equation
'           form ax2 + bx + c
'  inputs:   trendline equation
'  Outputs: firstpart of trendline equation
'------------------------------------------------------------------------
Function firstPartPolyTrendEq(ByVal trendlineEq As String) As Double

    tempStorage = Sheets(dataSheet).Cells(1, 1)
    Sheets(dataSheet).Cells(1, 1) = trendlineEq
```

```
        i = 1
        While found = False
           testchar = Sheets(dataSheet).Cells(1, 1).Characters(i, 1).Text
           If testchar = "x" Then
              found = True
           Else
           i = i + 1
           End If
          Wend

         'i = location of x
         'firstpart = starts at character 5
         'num of characters = location(x) - 5

         firstPartPolyTrendEq = Sheets(dataSheet).Cells(1, 1).Characters(5, i - 5).Text
         Sheets(dataSheet).Cells(1, 1) = tempStorage

     End Function ' first part poly order - 2 trendline
        '-------------------------------------------------------------------------
        '  Function: secondPartPolyTrendEq
        '  Author: Matt Behnke
        '  Created: 2/1/02
        '  Description: extracts the second part of a second order polygonal trendline the given trendline
equation
        '  inputs:   trendline equation
        '  Outputs: firstpart of trendline equation
        '-------------------------------------------------------------------------
     Function secondPartPolyTrendEq(ByVal trendlineEq As String) As Double

        tempStorage = Sheets(dataSheet).Cells(1, 1)
        Sheets(dataSheet).Cells(1, 1) = trendlineEq

        i = 1
        While found = False
           testchar = Sheets(dataSheet).Cells(1, 1).Characters(i, 1).Text
           If testchar = "x" Then
              found = True
           Else
           i = i + 1
           End If
          Wend

        j = i + 1
        While found2 = False
           testchar = Sheets(dataSheet).Cells(1, 1).Characters(j, 1).Text
           If testchar = "x" Then
              found2 = True
           Else
           j = j + 1
```

367

```
      End If
     Wend


     'i = location of first x
     'j = location of second x
     'secondpart starts at character i plus 5..
     '    i12345   j1234
     '1.0000x2 + 2.001x + 8.878
     'num of characters = j - i + 5


     secondPartPolyTrendEq = Sheets(dataSheet).Cells(1, 1).Characters(i + 5, j - (i + 5)).Text
     Sheets(dataSheet).Cells(1, 1) = tempStorage


   End Function ' secondPart poly eq


     '--------------------------------------------------------------------------
     ' Function: thirdPartPolyTrendEq
     ' Author: Matt Behnke
     ' Created: 2/1/02
     ' Description: extracts the second part of a second order polygonal trendline the given trendline
equation
     ' inputs:   trendline equation
     ' Outputs: firstpart of trendline equation
     '--------------------------------------------------------------------------
   Function thirdPartPolyTrendEq(ByVal trendlineEq As String) As Double


     tempStorage = Sheets(dataSheet).Cells(1, 1)
     Sheets(dataSheet).Cells(1, 1) = trendlineEq


     i = 1
     While found = False
       testchar = Sheets(dataSheet).Cells(1, 1).Characters(i, 1).Text
       If testchar = "x" Then
          found = True
       Else
        i = i + 1
        End If
      Wend


     j = i + 1
     While found2 = False
       testchar = Sheets(dataSheet).Cells(1, 1).Characters(j, 1).Text
       If testchar = "x" Then
          found2 = True
       Else
        j = j + 1
        End If
      Wend
```

368

```
'i = location of first x
'j = location of second x
'secondpart starts at character i plus 5..
'     i12345   j1234
'1.0000x2 + 2.001x + 8.878

'third part starts at character j plus 4..
'extract 5 characters..

thirdPartPolyTrendEq = Sheets(dataSheet).Cells(1, 1).Characters(j + 4, 5).Text
Sheets(dataSheet).Cells(1, 1) = tempStorage

End Function ' thirdPart poly eq


'--------------------------------------------------------------------------
'  Function: cols
'  Author: Matt Behnke
'  Created: 9/11/01
'  Description: changes column number into a letter.
'  inputs:   columnNumber
'  Outputs: column letter
'--------------------------------------------------------------------------
Function col(ByVal columnNumber As Integer) As String

  Select Case columnNumber
    Case 1
       col = "A"
    Case 2
       col = "B"
    Case 3
       col = "C"
    Case 4
       col = "D"
    Case 5
       col = "E"
    Case 6
       col = "F"
    Case 7
       col = "G"
    Case 8
       col = "H"
    Case 9
       col = "I"
    Case 10
       col = "J"
    Case 11
       col = "K"
    Case 12
       col = "L"
    Case 13
```

```
      col = "M"
Case 14
    col = "N"
Case 15
    col = "O"
Case 16
    col = "P"
Case 17
    col = "Q"
Case 18
    col = "R"
Case 19
    col = "S"
Case 20
    col = "T"
Case 21
    col = "U"
Case 22
    col = "V"
Case 23
    col = "W"
Case 24
    col = "X"
Case 25
    col = "Y"
Case 26
    col = "Z"
Case 27
    col = "AA"
Case 28
    col = "AB"
Case 29
    col = "AC"
Case 30
    col = "AD"
Case 31
    col = "AE"
Case 32
    col = "AF"
Case 33
    col = "AG"
Case 34
    col = "AH"
Case 35
    col = "AI"
Case 36
    col = "AJ"
Case 37
    col = "AK"
Case 38
```

```
      col = "AL"
Case 39
   col = "AM"
Case 40
   col = "AN"
Case 41
   col = "AO"
Case 42
   col = "AP"
Case 43
   col = "AQ"
Case 44
   col = "AR"
Case 45
   col = "AS"
Case 46
   col = "AT"
Case 47
   col = "AU"
Case 48
   col = "AV"
Case 49
   col = "AW"
Case 50
   col = "AX"
Case 51
   col = "AY"
Case 52
   col = "AZ"
Case 53
   col = "BA"
Case 54
   col = "BB"
Case 55
   col = "BC"
Case 56
   col = "BD"
Case 57
   col = "BE"
Case 58
   col = "BF"
Case 59
   col = "BG"
Case 60
   col = "BH"
Case 61
   col = "BI"
Case 62
   col = "BJ"
Case 63
```

```
        col = "BK"
Case 64
    col = "BL"
Case 65
    col = "BM"
Case 66
    col = "BN"
Case 67
    col = "BO"
Case 68
    col = "BP"
Case 69
    col = "BQ"
Case 70
    col = "BR"
Case 71
    col = "BS"
Case 72
    col = "BT"
Case 73
    col = "BU"
Case 74
    col = "BV"
Case 75
    col = "BW"
Case 76
    col = "BX"
Case 77
    col = "BY"
Case 78
    col = "BZ"
Case 79
    col = "CA"
Case 80
    col = "CB"
Case 81
    col = "CC"
Case 82
    col = "CD"
Case 83
    col = "CE"
Case 84
    col = "CF"
Case 85
    col = "CG"
Case 86
    col = "CH"
Case 87
    col = "CI"
Case 88
```

```
        col = "CJ"
Case 89
    col = "CK"
Case 90
    col = "CL"
Case 91
    col = "CM"
Case 92
    col = "CN"
Case 93
    col = "CO"
Case 94
    col = "CP"
Case 95
    col = "CQ"
Case 96
    col = "CR"
Case 97
    col = "CS"
Case 98
    col = "CT"
Case 99
    col = "CU"
Case 100
    col = "CV"
Case 101
    col = "CW"
Case 102
    col = "CX"
Case 103
    col = "CY"
Case 104
    col = "CZ"
Case 105
    col = "DA"
Case 106
    col = "DB"
Case 107
    col = "DC"
Case 108
    col = "DD"
Case 109
    col = "DE"
Case 110
    col = "DF"
Case 111
    col = "DG"
Case 112
    col = "DH"
Case 113
```

```
        col = "DI"
Case 114
   col = "DJ"
Case 115
   col = "DK"
Case 116
   col = "DL"
Case 117
   col = "DM"
Case 118
   col = "DN"
Case 119
   col = "DO"
Case 120
   col = "DP"
Case 121
   col = "DQ"
Case 122
   col = "DR"
Case 123
   col = "DS"
Case 124
   col = "DT"
Case 125
   col = "DU"
Case 126
   col = "DV"
Case 127
   col = "DW"
Case 128
   col = "DX"
Case 129
   col = "DY"
Case 130
   col = "DZ"
Case 131
   col = "EA"
Case 132
   col = "EB"
Case 133
   col = "EC"
Case 134
   col = "ED"
Case 135
   col = "EE"
Case 136
   col = "EF"
Case 137
   col = "EG"
Case 138
```

```
        col = "EH"
Case 139
  col = "EI"
Case 140
  col = "EJ"
Case 141
  col = "EK"
Case 142
  col = "EL"
Case 143
  col = "EM"
Case 144
  col = "EN"
Case 145
  col = "EO"
Case 146
  col = "EP"
Case 147
  col = "EQ"
Case 148
  col = "ER"
Case 149
  col = "ES"
Case 150
  col = "ET"
Case 151
  col = "EU"
Case 152
  col = "EV"
Case 153
  col = "EW"
Case 154
  col = "EX"
Case 155
  col = "EY"
Case 156
  col = "EZ"
Case 157
  col = "FA"
Case 158
  col = "FB"
Case 159
  col = "FC"
Case 160
  col = "FD"
Case 161
  col = "FE"
Case 162
  col = "FF"
Case 163
```

```
        col = "FG"
Case 164
  col = "FH"
Case 165
  col = "FI"
Case 166
  col = "FJ"
Case 167
  col = "FK"
Case 168
  col = "FL"
Case 169
  col = "FM"
Case 170
  col = "FN"
Case 171
  col = "FO"
Case 172
  col = "FP"
Case 173
  col = "FQ"
Case 174
  col = "FR"
Case 175
  col = "FS"
Case 176
  col = "FT"
Case 177
  col = "FU"
Case 178
  col = "FV"
Case 179
  col = "FW"
Case 180
  col = "FX"
Case 181
  col = "FY"
Case 182
  col = "FZ"
Case 183
  col = "GA"
Case 184
  col = "GB"
Case 185
  col = "GC"
Case 186
  col = "GD"
Case 187
  col = "GE"
Case 188
```

```
          col = "GF"
Case 189
   col = "GG"
Case 190
   col = "GH"
Case 191
   col = "GI"
Case 192
   col = "GJ"
Case 193
   col = "GK"
Case 194
   col = "GL"
Case 195
   col = "GM"
Case 196
   col = "GN"
Case 197
   col = "GO"
Case 198
   col = "GP"
Case 199
   col = "GQ"
Case 200
   col = "GR"
Case 201
   col = "GS"
Case 202
   col = "GT"
Case 203
   col = "GU"
Case 204
   col = "GV"
Case 205
   col = "GW"
Case 206
   col = "GX"
Case 207
   col = "GY"
Case 208
   col = "GZ"
Case 209
   col = "HA"
Case 210
   col = "HB"
Case 211
   col = "HC"
Case 212
   col = "HD"
Case 213
```

```
      col = "HE"
Case 214
    col = "HF"
Case 215
    col = "HG"
Case 216
    col = "HH"
Case 217
    col = "HI"
Case 218
    col = "HJ"
Case 219
    col = "HK"
Case 220
    col = "HL"
Case 221
    col = "HM"
Case 222
    col = "HN"
Case 223
    col = "HO"
Case 224
    col = "HP"
Case 225
    col = "HQ"
Case 226
    col = "HR"
Case 227
    col = "HS"
Case 228
    col = "HT"
Case 229
    col = "HU"
Case 230
    col = "HV"
Case 231
    col = "HW"
Case 232
    col = "HX"
Case 233
    col = "HY"
Case 234
    col = "HZ"
Case 235
    col = "IA"
Case 236
    col = "IB"
Case 237
    col = "IC"
Case 238
```

```vba
          col = "ID"
    Case 239
        col = "IE"
    Case 240
        col = "IF"
    Case 241
        col = "IG"
    Case 242
        col = "IH"
    Case 243
        col = "II"
    Case 244
        col = "IJ"
    Case 245
        col = "IK"
    Case 246
        col = "IL"
    Case 247
        col = "IM"
    Case 248
        col = "IN"
    Case 249
        col = "IO"
    Case 250
        col = "IP"
    Case 251
        col = "IQ"
    Case 252
        col = "IR"
    Case 253
        col = "IS"
    Case 254
        col = "IT"
    Case 255
        col = "IU"

    Case others
        col = "Z"
End Select

End Function 'col


Public Function Update_Mathcad_Band_Stats(ByVal mathcad_sheet_name As String, _
    ByVal data_sheet_name As String, ByVal start_cell_x As Variant, _
    ByVal start_cell_y As Variant, ByVal num_rows As Integer, ByVal fit_type As Integer, _
    ByVal tolerance As Double) As Variant
'-------------------------------------------------------------------------
' Function: Update_Mathcad_Band_Stats
' Author: Aaron Micyus
```

```
' Last Modified: 12/05/2001
' Description: Given location information for input data this subroutine
'  passes data to embedded mathcad object for processing and returns
'  obtained values
' inputs:
'     mathcad_sheet_name :   This is the name of the sheet the embedded
'                    object is in
'     data_sheet_name :   This is the name of the sheet we will obtain data
'                from
'     start_cell_x   :   This is the cell location we start getting x data from
'     start_cell_y   :   This is the cell location we start getting y data from
'     num_rows       :    This is the number of data rows we have
'     fit_type       :  integer value corresponding to fit type to return
'                   0 - Hyperbolic 3 parameter (k,p,r)
'                   1 - Exponential 3 parameter (k,p,r)
'                   2 - Power 2 parameter (b,m)
'     tolerance     :   tolerance value for fit
'
' Outputs:
'     array        :   returned array will hold calculated k,p,r values
'                   [1] element one : k
'                   [2] element two : r
'                   [3] element three: p
'                   [4] element four : Rsquared value
'-------------------------------------------------------------------------


    'VARS
    Dim Mathcad As Object                    'our interface to the Mathcad
                            'embedded object
    Dim data_x_real, data_x_imag As Variant      'vars for real and imaginary
                            'components of x data
    Dim data_y_real, data_y_imag As Variant      'vars for real and imaginary
                            'components of y data
    Dim tolerance_real, tolerance_imag As Variant   'vars for real and imaginary
                            'components of tolerance

    Dim k_real, k_imag As Variant                'vars for real and imag
                            'k values from mathcad
    Dim r_real, r_imag As Variant                'vars for real and imag
                            'r values from mathcad
    Dim p_real, p_imag As Variant                'vars for real and imag
                            'p values from mathcad
    Dim b_real, b_imag As Variant                'vars for real and imag
                            'b values from mathcad
    Dim m_real, m_imag As Variant                'vars for real and imag
                            'm values from mathcad
    Dim rsquared_real, rsquared_imag As Variant     'vars for real and imag
                            'r squared values from mathcad
```

380

```vba
        Dim current_char_position As Variant        'temp var to hold position in string
        Dim range_x, range_y As Variant             'vars for calculated ranges

        Dim fit_results(4) As Variant               'array to hold returned fit data from mathcad



        'initialize embedded mathcad
        Call Register_Mathcad_OLE(mathcad_sheet_name)

        'activate the sheet with the embedded mathcad object
        Worksheets(mathcad_sheet_name).Activate

        'get object reference
        Set Mathcad = ActiveSheet.OLEObjects(1).Object

        'activate the sheet with the data
        Worksheets(data_sheet_name).Activate

        """""'construct the x value range
        'this temp variable holds current position in string we are parsing through
        current_char_position = 1

        'traverse the string until we find a numeric character
        While Not IsNumeric(Mid(start_cell_x, current_char_position, 1))

           current_char_position = current_char_position + 1

        Wend

        'calculate range string for x
        range_x = start_cell_x & ":" & Left(start_cell_x, 1)
        range_x = range_x & (Right(start_cell_x, (Len(start_cell_x) - current_char_position + 1)) +
num_rows - 1)

        """"""'now construct y value range
        'this temp variable holds current position in string we are parsing through
        current_char_position = 1

        'traverse the string until we find a numeric character
        While Not IsNumeric(Mid(start_cell_y, current_char_position, 1))

           current_char_position = current_char_position + 1

        Wend

        'calculate range string for y
        range_y = start_cell_y & ":" & Left(start_cell_y, 1)
        range_y = range_y & (Right(start_cell_y, (Len(start_cell_y) - current_char_position + 1)) +
num_rows - 1)
```

```
'set ranges for input data for mathcad
data_x_real = ActiveSheet.Range(range_x).Value
data_x_imag = Empty
data_y_real = ActiveSheet.Range(range_y).Value
data_y_imag = Empty


tolerance_real = tolerance          'obtained from parameter
tolerance_imag = Empty

'import values into mathcad
Call Mathcad.SetComplex("X_in", data_x_real, data_x_imag)
Call Mathcad.SetComplex("Y_in", data_y_real, data_y_imag)
Call Mathcad.SetComplex("eTOL", tolerance_real, tolerance_imag)

'have mathcad recalculate sheet
Call Mathcad.Recalculate

If fit_type = HYP3_FIT Then

    'get values from mathcad for excel
    Call Mathcad.GetComplex("out0", k_real, k_imag)
    Call Mathcad.GetComplex("out1", r_real, r_imag)
    Call Mathcad.GetComplex("out2", p_real, p_imag)
    Call Mathcad.GetComplex("out3", rsquared_real, rsquared_imag)

    'fill array with results
    fit_results(1) = k_real
    fit_results(2) = r_real
    fit_results(3) = p_real
    fit_results(4) = rsquared_real

ElseIf fit_type = EXP3_FIT Then

    'get values from mathcad for excel
    Call Mathcad.GetComplex("out4", k_real, k_imag)
    Call Mathcad.GetComplex("out5", r_real, r_imag)
    Call Mathcad.GetComplex("out6", p_real, p_imag)
    Call Mathcad.GetComplex("out7", rsquared_real, rsquared_imag)

    'fill array with results
    fit_results(1) = k_real
    fit_results(2) = r_real
    fit_results(3) = p_real
    fit_results(4) = rsquared_real

ElseIf fit_type = POW2_FIT Then

    'get values from mathcad for excel
```

```
        Call Mathcad.GetComplex("out8", b_real, b_imag)
        Call Mathcad.GetComplex("out9", m_real, m_imag)
        Call Mathcad.GetComplex("out10", rsquared_real, rsquared_imag)

        'fill array with results
        fit_results(1) = b_real
        fit_results(2) = m_real
        fit_results(3) = Empty
        fit_results(4) = rsquared_real

    End If

    Update_Mathcad_Band_Stats = fit_results

'end of Update_Mathcad_Band_Stats
End Function

Public Function Register_Mathcad_OLE(ByVal mathcad_sheet_name As String)

'
' register_mathcad_ole Macro
' opens embedded mathcad document in order for system to recognize it for future macro
'

'

    Sheets(mathcad_sheet_name).Select
    Range("A1").Activate
    ActiveSheet.Shapes("Object 1").Select
    Selection.Verb Verb:=xlPrimary
    Range("A1").Activate
    'Sheets("A_Band_Stats").Select
End Function

'------------------------------------------------------------------------
'  sub: fillMonthsCol
'  Author: Matt Behnke
'  Created: 12/11/01
'  Description: fills in the months if they are missing.. Inserts a column used for a matrix sheet
'           this doesnt work because there are not enough columns
'  inputs:  sheetName
'  Outputs:
'------------------------------------------------------------------------
Sub fillMonthsCol() ' ByVal sheetName As String)

    sheetName = ActiveSheet.Name
    numColumns = CountCols(sheetName, 1)
    Dim theMonth As Integer

    counter = 1
    monthCounter = "" & counter
```

```
For i = 4 To numColumns

  j = 1
  While found = False
    testchar = Cells(1, i).Characters(j, 1).Text
    If testchar = "/" Then
      found = True
    Else
    j = j + 1
    End If
  Wend
  'month ends at j

  currentMonth = Cells(1, i).Characters(1, j - 1).Text
  If currentMonth < 10 Then
    restofDate = Cells(1, i).Characters(2, 5).Text
  Else
    restofDate = Cells(1, i).Characters(3, 5).Text
  End If
  theMonth = currentMonth
  If theMonth > monthCounter Then
    While theMonth > monthCounter
      Range(Cells(1, i), Cells(1, i)).Select
      Selection.EntireColumn.Insert

      'copy previous column
      Columns(col(i - 1) & ":" & col(i - 1)).Select
      Selection.Copy
      Columns(col(i) & ":" & col(i)).Select
      ActiveSheet.Paste
      Cells(1, i) = "" & monthCounter & restofDate

      counter = counter + 1
      If counter = 13 Then
        counter = 1
      End If
      monthCounter = "" & counter
      i = i + 1
      numColumns = numColumns + 1
    Wend
  End If


  counter = counter + 1
  If counter = 13 Then
    counter = 1
  End If
  monthCounter = "" & counter
  found = False
Next i
```

384

```vba
      Columns("D:D").ColumnWidth = 6.29
End Sub ' fill months col


'---------------------------------------------------------------------------
' sub: qLevelSummary
' Author: Matt Behnke
' Created: 1/14/02
' Description: creates a summary sheet for a q_level, lists time steps and num of instances
'              for each q level
' inputs:  qlevelType: author or term
'          prefix: the sheet prefix, changes whether its author or term
'          numqLevels: the number of q levels
' Outputs:
'---------------------------------------------------------------------------
Sub qLevelSummary(ByVal qLevelType As String, ByVal prefix As String, ByVal numqLevels As Integer)

For z = 1 To numqLevels
   If z < 10 Then
      sheetName = prefix & "0" & z & "_" & qLevelType & "_month"
      summarySheet = prefix & "0" & z & "_" & qLevelType & "_summary"
   Else
      sheetName = prefix & "" & z & "_" & qLevelType & "_month"
      summarySheet = prefix & "" & z & "_" & qLevelType & "_summary"
   End If

   Sheets.Add After:=Sheets(Sheets.Count)
   Sheets(Sheets.Count).Select
   ActiveSheet.Name = summarySheet

   numColumns = CountCols(sheetName, 1)
   numRows = CountRows(sheetName, 1)

   Sheets(summarySheet).Cells(1, 1) = "q level"
   Sheets(summarySheet).Cells(1, 2) = z


   Sheets(summarySheet).Cells(2, 1) = " "
   Sheets(summarySheet).Cells(3, 1) = "Time Steps"
   Sheets(summarySheet).Cells(3, 2) = "sum"
   Sheets(summarySheet).Cells(3, 3) = "count"

   For i = 4 To numColumns
      Sheets(summarySheet).Cells(i, 1) = Sheets(sheetName).Cells(1, i)
      Sheets(summarySheet).Cells(i, 2) = "=SUM('" & sheetName & "'!" & col(i) & "2:" & col(i)
& numRows & ")"
      Sheets(summarySheet).Cells(i, 3) = "=Count('" & sheetName & "'!" & col(i) & "2:" & col(i)
& numRows & ")"
   Next i
Next z
```

End Sub

```
'------------------------------------------------------------------------
'  sub: qLevelTrigger
'  Author: Matt Behnke
'  Created: 1/18/02
'  Description: activates the qlevel functions
'  inputs:
'  Outputs:
'------------------------------------------------------------------------
Sub qLevelTrigger()

qLevelType = InputBox("Enter Author or Term:")
numqLevels = InputBox("Enter number of qLevels:")

If qLevelType = "Author" Or qLevelType = "Term" And numqLevels > 0 Then 'check input

   If qLevelType = "Author" Then
      prefix = "_"
   Else
      prefix = ""
   End If

   'Call qLevelCumulative(qLevelType, prefix, numqLevels)
'    Call qLevelSummary(qLevelType, prefix, numqLevels)
   'Call qLevelYears(qLevelType, prefix, numqLevels)
   ' Call qLevelMonths(qLevelType, prefix, numqLevels, False)
   'Call qLevelMonths(qLevelType, prefix, numqLevels, True) 'calculate mass
   ' Call qLevelMonthsCount(qLevelType, prefix, numqLevels)
   Call qLevelEntropy(qLevelType, prefix, numqLevels)
   'Call qLevelEntropy2(numqLevels) 'wrong!!!!!!!!!!!!!!
   'Call qLevelMonthTemp(numqLevels)

Else
   MsgBox ("WRONG INPUT.. TRY AGAIN!")
End If

End Sub

'------------------------------------------------------------------------
'  sub: qLevelCumulative
'  Author: Matt Behnke
'  Created: 1/16/02
'  Description: ..puts in cumulative values for each time step, and sums at the bottom
'  inputs:  qlevelType: author or term
'          prefix: the sheet prefix, changes whether its author or term
'          numqLevels: the number of q levels
'  Outputs:
'------------------------------------------------------------------------
```

```vba
        Sub qLevelCumulative(ByVal qLevelType As String, ByVal prefix As String, ByVal numqLevels
As Integer)

        For i = 1 To numqLevels

           If i < 10 Then
              Call CalcCumulative(prefix & "0" & i & "_" & qLevelType & "_month")
           Else
              Call CalcCumulative(prefix & "" & i & "_" & qLevelType & "_month")
           End If
        Next i

        End Sub

        '-------------------------------------------------------------------------
        ' sub: qLevelYears
        ' Author: Matt Behnke
        ' Created: 1/16/02
        ' Description: uses an array of years to store the amount of instances for a year.
        '          outputs each year of the dataset to each summary sheet and number of instances
        ' inputs:  qlevelType: author or term
        '          prefix: the sheet prefix, changes whether its author or term
        '          numqLevels: the number of q levels
        ' Outputs:
        '-------------------------------------------------------------------------
        Sub qLevelYears(ByVal qLevelType As String, ByVal prefix As String, ByVal numqLevels As
Integer)

        'number of ntuples
        ntuples = numqLevels

        Dim years As Variant

        overallQSummary = "q_summary_year"

        firstYear = "2500"
        firstYearOffset = 6
        lastYear = 0

        Sheets.Add After:=Sheets(Sheets.Count)
        Sheets(Sheets.Count).Select
        ActiveSheet.Name = overallQSummary

        For z = 1 To ntuples

           years = Array(0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0, 0, 0)

           If z < 10 Then
              summarySheet = prefix & "0" & z & "_" & qLevelType & "_summary"
```

```
      Else
         summarySheet = prefix & "" & z & "_" & qLevelType & "_summary"
      End If

      numRows = CountRows(summarySheet, 1)

      For i = 4 To numRows

         'determine current year
         j = 1
         While found = False
            testchar = Cells(i, 1).Characters(j, 1).Text
            If testchar = "/" Then
               found = True
            Else
            j = j + 1
            End If
         Wend
         'month ends at j

         currentMonth = Cells(i, 1).Characters(1, j - 1).Text
         If currentMonth < 10 Then
            theYear = Cells(i, 1).Characters(5, 2).Text
         Else
            theYear = Cells(i, 1).Characters(6, 2).Text
         End If

         If theYear > 50 Then 'add prefix to the year
            theYear = "19" & theYear
         Else
            theYear = "20" & theYear
         End If

         'check to see if the currentYear is less than first year
         If theYear < firstYear Then
            firstYear = theYear
            firstYearOffset = firstYearOffset - 1 'array index
         End If

         If theYear > lastYear Then
            lastYear = theYear
         End If

         yearOffset = theYear - firstYear + 5  'so there can be 5 years less data than the first firstYear
value
         years(yearOffset) = Sheets(summarySheet).Cells(i, 2)
         found = False
      Next i

      'output yearsArray
```

```
              Sheets(summarySheet).Cells(3, 4) = "Years"
              Sheets(summarySheet).Cells(3, 5) = "instances"

              counter = 4 'for output


              For x = firstYearOffset To lastYear - firstYear + firstYearOffset
                 Sheets(summarySheet).Cells(counter, 4) = firstYear + x - firstYearOffset
                 Sheets(summarySheet).Cells(counter, 5) = years(x) 'instances
                 If z = 1 Then
                    Sheets(overallQSummary).Cells(counter, z) = firstYear + x - firstYearOffset
                    Sheets(overallQSummary).Cells(counter, z + 1) = 0
                 End If
                 Sheets(overallQSummary).Cells(counter, z + 2) = years(x)

                 'if there is a zero in a year then the put the previous years value into the current year
(cumulative)
                 If Sheets(overallQSummary).Cells(counter, z + 2) = 0 And x > firstYearOffset Then
                    Sheets(overallQSummary).Cells(counter, z + 2) = Sheets(overallQSummary).Cells(counter
- 1, z + 2)
                 End If

                 If x = firstYearOffset Then
                    Sheets(overallQSummary).Cells(counter - 1, z + 2) = z
                 End If


              counter = counter + 1

              Next x

           Next z
           'copy chart

              currFilename = Application.ActiveWorkbook.Name
           'FIX
THIS********************************************************************************
********************:
              Windows("AffiliationMacro3.xls").Activate
              Sheets("q_level_yr").Select
              Sheets("q_level_yr").Copy After:=Workbooks(currFilename).Sheets(1)

              counter = 4
              columnStart = 2
              For z = 2 To ntuples
                 ActiveChart.SeriesCollection(z - 1).Values = "=" & overallQSummary & "!R" & counter &
"C" & columnStart & ":R" & counter & "C" & ntuples
                 counter = counter + 1
              Next z
```

```
            End Sub

            '----------------------------------------------------------------------
            '  sub: qLevelMonths
            '  Author: Matt Behnke
            '  Created: 1/27/02 --------finished 2/4/02
            '  Description: uses an array of years and months to store the amount of instances for each timestep
k.
            '            outputs the number of cumulative instances per month / year
            '  inputs:  qlevelType: author or term
            '           prefix: the sheet prefix, changes whether its author or term
            '           numqLevels: the number of q levels
            '  Outputs:
            '----------------------------------------------------------------------
            Sub qLevelMonths(ByVal qLevelType As String, ByVal prefix As String, ByVal numqLevels As
Integer, ByVal mass As Boolean)

            'number of ntuples
            ntuples = numqLevels

            Dim years As Variant
            Dim yearsMonths As Variant

            firstYear = "2500"
            lastYear = 0

            If mass = True Then
               overallQSummary = "q_summary_monthly_count_mass"
            Else
               overallQSummary = "q_summary_monthly_count"
            End If

            Sheets.Add After:=Sheets(Sheets.Count)
            Sheets(Sheets.Count).Select
            ActiveSheet.Name = overallQSummary

            Sheets(overallQSummary).Cells(1, 1) = "  "
            Sheets(overallQSummary).Cells(2, 1) = "  "
            Sheets(overallQSummary).Cells(3, 1) = "  "

            Cells(4, 2).Select
            ActiveWindow.FreezePanes = True

            years = Array()
            yearsMonths = Array()
            'ReDim yearsMonths(0 To 0, 1 To 12)

            For z = 1 To ntuples
```

```
'get the source sheet's name
If z < 10 Then
   summarySheet = prefix & "0" & z & "_" & qLevelType & "_summary"
Else
   summarySheet = prefix & "" & z & "_" & qLevelType & "_summary"
End If

numRows = CountRows(summarySheet, 1)

'scan the first sheet to get the last and first year to get the array bounds
If z = 1 Then

   For i = 4 To numRows

      'determine current year
      j = 1
      While found = False
         testchar = Cells(i, 1).Characters(j, 1).Text
         If testchar = "/" Then
            found = True
         Else
         j = j + 1
         End If
      Wend
      'month ends at j

      currentMonth = Cells(i, 1).Characters(1, j - 1).Text
      If currentMonth < 10 Then
         theYear = Cells(i, 1).Characters(5, 2).Text
      Else
         theYear = Cells(i, 1).Characters(6, 2).Text
      End If

      If theYear > 50 Then 'add prefix to the year
         theYear = "19" & theYear
      Else
         theYear = "20" & theYear
      End If

      'check to see if the currentYear is less than first year
      If theYear < firstYear Then
         firstYear = theYear
      End If

      If theYear > lastYear Then
         lastYear = theYear
      End If

      found = False
   Next i 'done scanning the sheet now redim the array
```

```
        'ReDim yearsMonths(firstYear To lastYear, 1 To 12)

End If 'z = 1

ReDim yearsMonths(firstYear To lastYear, 1 To 12)

For i = 4 To numRows 'now process all the nTuple sheets

    'determine current year
    j = 1
    While found = False
        testchar = Cells(i, 1).Characters(j, 1).Text
        If testchar = "/" Then
            found = True
        Else
        j = j + 1
        End If
    Wend
    'month ends at j

    currentMonth = Cells(i, 1).Characters(1, j - 1).Text
    If currentMonth < 10 Then
        theYear = Cells(i, 1).Characters(5, 2).Text
    Else
        theYear = Cells(i, 1).Characters(6, 2).Text
    End If

    If theYear > 50 Then 'add prefix to the year
        theYear = "19" & theYear
    Else
        theYear = "20" & theYear
    End If

    yearsMonths(theYear, currentMonth) = Sheets(summarySheet).Cells(i, 2)
    found = False
Next i

'output yearsArray

'Sheets(summarySheet).Cells(3, 4) = "Years"
'Sheets(summarySheet).Cells(3, 5) = "instances"

counter = 4 'for output
monthCounter = 1
lastValue = 0


For j = firstYear To lastYear
```

```vba
        For k = 1 To 12

            If "" & j = firstYear And k = 1 Then 'label the q levels
                Sheets(overallQSummary).Cells(counter - 1, z + 2) = z
            End If

            If z = 1 Then 'put the month/year
                Sheets(overallQSummary).Cells(counter, z) = "" & monthCounter & "/" & j
                Sheets(overallQSummary).Cells(counter, z + 1) = 0
            End If

            currentValue = yearsMonths(j, monthCounter)
            If mass = True Then
                multiplyer = z
            Else
                multiplyer = 1
            End If

            If currentValue > lastValue Then
                Sheets(overallQSummary).Cells(counter, z + 2) = currentValue * multiplyer
                lastValue = currentValue
            Else
                Sheets(overallQSummary).Cells(counter, z + 2) = lastValue * multiplyer
            End If

            monthCounter = monthCounter + 1
            If monthCounter > 12 Then
                monthCounter = 1
            End If
            counter = counter + 1

        Next k
      Next j
      If z = ntuples Then
'         Sheets(overallQSummary).Cells(counter - 1, z + 3) = "=SUM(" * col(2) & counter - 1 _
'                              & ":" & col(nTuples + 2) & counter - 1 & ")"
      End If


    Next z
    'total of all the qlevels:
    Sheets(overallQSummary).Cells(1, 1) = "=SUM(" & col(2) & counter - 1 & ":" & col(ntuples + 2)
& counter - 1 & ")"



    'copy chart

'    currFilename = Application.ActiveWorkbook.Name
```

```
'   Windows("AffiliationMacro.xls").Activate
'   Sheets("q_level_yr").Select
'   Sheets("q_level_yr").Copy After:=Workbooks(currFilename).Sheets(Sheets.Count)

'   counter = 4
'   columnStart = 2
'   For z = 2 To nTuples
'       ActiveChart.SeriesCollection(z - 1).Values = "=" & overallQSummary & "!R" & counter &
"C" & columnStart & ":R" & counter & "C" & nTuples
'       counter = counter + 1
'   Next z

End Sub 'qlevel months


'------------------------------------------------------------------------
'   sub: qLevelMonthsCOUNT
'   Author: Matt Behnke
'   Created: 2/17/02
'   Description: uses an array of years and months to store the amount of instances for each timestep
k.
'               outputs the number of terms in the vocab per month / year puts
'   inputs:  qlevelType: author or term
'           prefix: the sheet prefix, changes whether its author or term
'           numqLevels: the number of q levels

'   Outputs:
'------------------------------------------------------------------------
Sub qLevelMonthsCount(ByVal qLevelType As String, ByVal prefix As String, ByVal
numqLevels As Integer)

'number of ntuples
ntuples = numqLevels

Dim years As Variant
Dim yearsMonths As Variant

firstYear = "2500"
lastYear = 0

    overallQSummary = "q_summary_monthly_count_count"

Sheets.Add After:=Sheets(Sheets.Count)
Sheets(Sheets.Count).Select
ActiveSheet.Name = overallQSummary

Sheets(overallQSummary).Cells(1, 1) = "  "
Sheets(overallQSummary).Cells(2, 1) = "  "
Sheets(overallQSummary).Cells(3, 1) = "  "

Cells(4, 2).Select
```

```vba
ActiveWindow.FreezePanes = True

years = Array()
yearsMonths = Array()
'ReDim yearsMonths(0 To 0, 1 To 12)

For z = 1 To ntuples

    'get the source sheet's name
    If z < 10 Then
        summarySheet = prefix & "0" & z & "_" & qLevelType & "_summary"
    Else
        summarySheet = prefix & "" & z & "_" & qLevelType & "_summary"
    End If

    numRows = CountRows(summarySheet, 1)

    'scan the first sheet to get the last and first year to get the array bounds
    If z = 1 Then

        For i = 4 To numRows

            'determine current year
            j = 1
            While found = False
                testchar = Cells(i, 1).Characters(j, 1).Text
                If testchar = "/" Then
                    found = True
                Else
                j = j + 1
                End If
            Wend
            'month ends at j

            currentMonth = Cells(i, 1).Characters(1, j - 1).Text
            If currentMonth < 10 Then
                theYear = Cells(i, 1).Characters(5, 2).Text
            Else
                theYear = Cells(i, 1).Characters(6, 2).Text
            End If

            If theYear > 50 Then 'add prefix to the year
                theYear = "19" & theYear
            Else
                theYear = "20" & theYear
            End If

            'check to see if the currentYear is less than first year
            If theYear < firstYear Then
                firstYear = theYear
```

```
      End If

    If theYear > lastYear Then
       lastYear = theYear
    End If

    found = False
  Next i 'done scanning the sheet now redim the array

  'ReDim yearsMonths(firstYear To lastYear, 1 To 12)

End If 'z = 1

ReDim yearsMonths(firstYear To lastYear, 1 To 12)

For i = 4 To numRows 'now process all the nTuple sheets

  'determine current year
  j = 1
  While found = False
     testchar = Cells(i, 1).Characters(j, 1).Text
     If testchar = "/" Then
        found = True
     Else
     j = j + 1
     End If
  Wend
  'month ends at j

  currentMonth = Cells(i, 1).Characters(1, j - 1).Text
  If currentMonth < 10 Then
     theYear = Cells(i, 1).Characters(5, 2).Text
  Else
     theYear = Cells(i, 1).Characters(6, 2).Text
  End If

  If theYear > 50 Then 'add prefix to the year
     theYear = "19" & theYear
  Else
     theYear = "20" & theYear
  End If

  yearsMonths(theYear, currentMonth) = Sheets(summarySheet).Cells(i, 3) 'count
  found = False
Next i

counter = 4 'for output
monthCounter = 1
lastValue = 0
```

```vba
        'numYears                =                UBound(yearsMonthsa)
'**************************************************WORK ON THIS later!!

          For j = firstYear To lastYear

            For k = 1 To 12

              If "" & j = firstYear And k = 1 Then 'label the q levels
                 Sheets(overallQSummary).Cells(counter - 1, z + 2) = z
              End If

              If z = 1 Then 'put the month/year
                 Sheets(overallQSummary).Cells(counter, z) = "" & monthCounter & "/" & j
                 Sheets(overallQSummary).Cells(counter, z + 1) = 0
              End If

              currentValue = yearsMonths(j, monthCounter)
              If mass = True Then
                 multiplyer = z
              Else
                 multiplyer = 1
              End If

              If currentValue > lastValue Then
                 Sheets(overallQSummary).Cells(counter, z + 2) = currentValue * multiplyer
                 lastValue = currentValue
              Else
                 Sheets(overallQSummary).Cells(counter, z + 2) = lastValue * multiplyer
              End If

              monthCounter = monthCounter + 1
              If monthCounter > 12 Then
                 monthCounter = 1
              End If
              counter = counter + 1

            Next k
          Next j
'    If z = nTuples Then
'       Sheets(overallQSummary).Cells(counter - 1, z + 3) = "=SUM(" * col(2) & counter - 1 _
'                        & ":" & col(nTuples + 2) & counter - 1 & ")"
'    End If


        Next z
        'total of all the qlevels:
        Sheets(overallQSummary).Cells(1, 1) = "=SUM(" & col(2) & counter - 1 & ":" & col(ntuples + 2)
& counter - 1 & ")"

        End Sub 'qlevel months CoUNT
```

```
'-----------------------------------------------------------------------
' sub: qLevelEntropy
' Author: Matt Behnke
' Created:  2/4/02
' finished: 3/11/02
' Description: uses an array of years and months to store the amount of entropy for each timestep
k.
'            outputs the cumulative entropy per timestep per q level
' inputs:  qlevelType: author or term
'          prefix: the sheet prefix, changes whether its author or term
'          numqLevels: the number of q levels
' Outputs:
'-----------------------------------------------------------------------
Sub qLevelEntropy(ByVal qLevelType As String, ByVal prefix As String, ByVal numqLevels As
Integer)

    'number of ntuples
    ntuples = numqLevels

    Dim years As Variant
    Dim yearsMonths As Variant
    Dim contributionEntropy As Variant

    firstYear = "2500"
    lastYear = 0

    overallQEntropy = "q_local_entropy_monthly"
    contributionEntropySheet = "q_contribution_entropy_monthly"
    countSheet = "q_summary_monthly_count"

    'add the contribution entropy sheet
    Sheets.Add After:=Sheets(Sheets.Count)
    Sheets(Sheets.Count).Select
    ActiveSheet.Name = contributionEntropySheet
    Cells(4, 2).Select
    ActiveWindow.FreezePanes = True

    'add the local entropy sheet
    Sheets.Add After:=Sheets(Sheets.Count)
    Sheets(Sheets.Count).Select
    ActiveSheet.Name = overallQEntropy
    Cells(4, 2).Select
    ActiveWindow.FreezePanes = True

    years = Array()
    yearsMonths = Array()
    contributionEntropy = Array()

    'ReDim yearsMonths(0 To 0, 1 To 12)
```

398

```
For z = 1 To ntuples

    'get the source sheet's name
    If z < 10 Then
        summarySheet = prefix & "0" & z & "_" & qLevelType & "_month"
    Else
        summarySheet = prefix & "" & z & "_" & qLevelType & "_month"
    End If

    numRows = CountRows(summarySheet, 1)
    numCols = CountCols(summarySheet, 1)

    'find the row that contains the time counts for the current time step...

    'scan the first sheet to get the last and first year to get the array bounds
    If z = 1 Then

        For i = 4 To numCols

            'determine current year
            j = 1
            While found = False
                testchar = Cells(1, i).Characters(j, 1).Text
                If testchar = "/" Then
                    found = True
                Else
                j = j + 1
                End If
            Wend
            'month ends at j

            currentMonth = Cells(1, i).Characters(1, j - 1).Text
            If currentMonth < 10 Then
                theYear = Cells(1, i).Characters(5, 2).Text
            Else
                theYear = Cells(1, i).Characters(6, 2).Text
            End If

            If theYear > 50 Then 'add prefix to the year
                theYear = "19" & theYear
            Else
                theYear = "20" & theYear
            End If

            'check to see if the currentYear is less than first year
            If theYear < firstYear Then
                firstYear = theYear
            End If
```

```
      If theYear > lastYear Then
         lastYear = theYear
      End If

      found = False
   Next i 'done scanning the sheet now redim the array

   'ReDim yearsMonths(firstYear To lastYear, 1 To 12)

End If 'z = 1

ReDim yearsMonths(firstYear To lastYear, 1 To 12)
ReDim contributionEntropy(firstYear To lastYear, 1 To 12)

For i = 4 To numCols 'now process all the nTuple sheets

   'determine current year
   j = 1
   While found = False
      testchar = Sheets(summarySheet).Cells(1, i).Characters(j, 1).Text
      If testchar = "/" Then
         found = True
      Else
      j = j + 1
      'found = True
      End If
   Wend
   'month ends at j

   currentMonth = Sheets(summarySheet).Cells(1, i).Characters(1, j - 1).Text

   If currentMonth < 10 Then
      theYear = Sheets(summarySheet).Cells(1, i).Characters(5, 2).Text
   Else
      theYear = Sheets(summarySheet).Cells(1, i).Characters(6, 2).Text
   End If

   If theYear > 50 Then 'add prefix to the year
      theYear = "19" & theYear
   Else
      theYear = "20" & theYear
   End If
   '*************

   timeStep = "" & currentMonth & "/" & theYear
   timeStepRow = findStringInSheet(countSheet, timeStep, "A")

 '  temp = Sheets(summarySheet).Cells(1, 1)
 '  Sheets(summarySheet).Cells(1, 1) = timeStepRange
 '  timeStepRow = Sheets(summarySheet).Cells(1, 1).Characters(4, 5).Text
```

```vba
'  Sheets(summarySheet).Cells(1, 1) = temp

        totalInstances = Sheets(countSheet).Cells(1, 1) 'timeStepRow, ntuples + 3)  'O_Q
        localSumInstances = Sheets(countSheet).Cells(timeStepRow, z + 2)          'O_q_k

        '
        ' Sh(A)_k_q = num instances A at k in q_level          num instances A at k in q_lvl
        '            -------------------------------- * log2 --------------------
        '            sum of instances at k in q_lvl          sum of instances at k in q_lvl


        '
        ' Sh(k)_q = Sh(A)_k + Sh(B)_k + ...
        '                     [ sum instances at k in q_lvl (O_q_k)       ]
        'contribution Cs_qlevel_k = abs [ -------------------------------------- ]  * Sh(k)_q
        '                     [ sum instances at all Q_lvls (O_Q)   ]
        '           O_q_k      O_Q
        '        + ----- * log2 -----
        '           O_Q        O_q_k
        '

        'traverse all the rows in the summarySheet to get the num of instances of each term
        'and the entropies of each term
        'store the sum of the entropies of each term in step k in the yearsMonths array.

        For j = 2 To numRows
          If Sheets(summarySheet).Cells(j, i) > 0 Then
            theValue = Sheets(summarySheet).Cells(j, i)
            entropy = (-theValue / localSumInstances) * (Log(theValue / localSumInstances) /
Log(2))

            yearsMonths(theYear, currentMonth) = yearsMonths(theYear, currentMonth) + entropy
          End If

          'at the last term in the time step compute the contribution entropy
          If j = numRows Then

            contributionEntropy(theYear, currentMonth) = Abs(localSumInstances / totalInstances)
_
                          *   yearsMonths(theYear,   currentMonth)   +   ((localSumInstances   /
totalInstances) _
                          * (Log(totalInstances / localSumInstances) / Log(2)))
          End If
        Next j

        found = False
     Next i

     'output yearsArray

     'Sheets(summarySheet).Cells(3, 4) = "Years"
```

```vba
'Sheets(summarySheet).Cells(3, 5) = "instances"

counter = 4 'for output
monthCounter = 1
lastValue = 0
lastContributionValue = 0

For j = firstYear To lastYear

   For k = 1 To 12

      If "" & j = firstYear And k = 1 Then 'label the q level
         Sheets(overallQEntropy).Cells(counter - 1, z + 2) = z
      End If

      If z = 1 Then 'put the month/year
         Sheets(overallQEntropy).Cells(counter, z) = "" & monthCounter & "/" & j
         Sheets(overallQEntropy).Cells(counter, z + 1) = 0

         Sheets("q_contribution_entropy_monthly").Cells(counter, z) = "" & monthCounter &
"/" & j

         Sheets("q_contribution_entropy_monthly").Cells(counter, z + 1) = 0
      End If

      currentValue = yearsMonths(j, monthCounter)
      currentContributionValue = contributionEntropy(j, monthCounter)
      If currentValue > lastValue Then
         Sheets(overallQEntropy).Cells(counter, z + 2) = currentValue

         lastValue = currentValue
      Else
         Sheets(overallQEntropy).Cells(counter, z + 2) = lastValue

      End If

      If currentContributionValue > lastContributionValue Then
         Sheets(contributionEntropySheet).Cells(counter, z + 2) = currentContributionValue
         lastContributionValue = currentContributionValue
      Else
         Sheets(contributionEntropySheet).Cells(counter, z + 2) = lastContributionValue
      End If

      monthCounter = monthCounter + 1
      If monthCounter > 12 Then
         monthCounter = 1
      End If
      counter = counter + 1

   Next k
Next j
```

```
        Next z
        'total of all the qlevels:
        'Sheets(overallQEntropy).Cells(1, 1) = "=SUM(" & col(2) & counter & ":" & col(nTuples + 2) &
counter

        End Sub ' qlevelEntropy




        '-------------------------------------------------------------------------
        ' sub: qLevelEntropy2  ---- OBSOLETE??
        ' Author: Matt Behnke
        ' Created:  2/4/02
        ' Description: uses an array of years and months to store the amount of entropy for each timestep
k.
        '            outputs the cumulative entropy per timestep per q level
        ' inputs: numQlevels
        ' Outputs:
        '-------------------------------------------------------------------------
        Sub qLevelEntropy2(ByVal numqLevels As Integer)


        'the source datasheet contains the timesteps and the count of instances in each q level per time
step..
        dataSheet = "q_summary_monthly_count"
        tempSheet = "q_level_monthly_Entropy"

        numRows = CountRows(dataSheet, 1)
        numValues = ntuples

        Sheets.Add After:=Sheets(Sheets.Count)
        Sheets(Sheets.Count).Select
        ActiveSheet.Name = tempSheet

        Cells(4, 2).Select
        ActiveWindow.FreezePanes = True

        'copy the sheet.. datasheet --> tempsheet
          Sheets(dataSheet).Select
          Cells.Select
          Selection.Copy
          Sheets(tempSheet).Select
          Range("A1").Select
          ActiveSheet.Paste


        totalInstances = Sheets(dataSheet).Cells(1, 1) 'THe total num of instances over the whole dataset
```

```
For k = 4 To numRows 'traverse all the steps

  For z = 1 To numqLevels 'traverse all the nTuples
       If Sheets(dataSheet).Cells(k, z + 2) > 0 Then
          theValue = Sheets(dataSheet).Cells(k, z + 2)
          entropy = (-theValue / totalInstances) * (Log(theValue / totalInstances) / Log(2))
          Sheets(tempSheet).Cells(k, z + 2) = entropy
       End If
  Next z

Next k



End Sub ' qlevelEntropy2

'----------------------------------------------------------------------------
'  sub: qLevelMonthTemp
'  Author: Matt Behnke
'  Created: 2/28/02
'  Description: uses the counts from the q level month sheet to:
'          1) store the instances of each time step into an array
'          2) calculates the probabilities of each instance
'          3) determines the x & y values
'          4) determines the alpha, beta values from the curvefit
'          5) outputs timesteps, and alpha and beta onto a new sheet
'          coeff(1) = A
'          coeff(2) = B
'          alpha = B
'          beta = e^(-A/alpha)
'  inputs:  numqLevels - the number of qlevels nTuples..
'  Outputs:
'----------------------------------------------------------------------------
Sub qLevelMonthTemp(ByVal numqLevels As Integer)

'number of ntuples
ntuples = numqLevels

Dim instances_q(64) As Variant            'stores the instances in each q_level q_i_k
Dim probabilities(64) As Variant          'stores the probabilities of instances P(q_i_k)
Dim x_values(64) As Variant               'x_values [X: ln(qi+r)]  -weibull
Dim y_values(64) As Variant               'y_values [Y: ln[-ln(1-P(qi)] -weibull
Dim coefficients As Variant               'coefficients
Dim numRows As Integer                    'number of rows on the datasheet
Dim numInstances_k As Double              'total num of instances at step k
Dim gamma As Double                       'the shift, r
Dim numValues As Integer                  'the number of values in the x,y arrays

gamma = 0#
```

404

'the source datasheet contains the timesteps and the count of instances in each q level per time step..
```
        dataSheet = "q_summary_monthly_count"
        tempSheet = "q_level_monthly_temp"

        numRows = CountRows(dataSheet, 1)
        numValues = ntuples

        Sheets.Add After:=Sheets(Sheets.Count)
        Sheets(Sheets.Count).Select
        ActiveSheet.Name = tempSheet

        Cells(4, 2).Select
        ActiveWindow.FreezePanes = True

        Sheets(tempSheet).Cells(1, 1) = " "
        Sheets(tempSheet).Cells(2, 1) = " "
        Sheets(tempSheet).Cells(3, 1) = "k"
        Sheets(tempSheet).Cells(3, 2) = "interval"
        Sheets(tempSheet).Cells(3, 3) = "A"
        Sheets(tempSheet).Cells(3, 4) = "B"
        Sheets(tempSheet).Cells(3, 5) = "alpha = B"
        Sheets(tempSheet).Cells(3, 6) = "beta = e^(-A/alpha)"
        Sheets(tempSheet).Cells(2, 6) = "T = beta"

        'numInstances_k = Sheets(dataSheet).Cells(1, 1) 'THe total num of instances over the whole dataset

        For k = 4 To numRows 'traverse all the steps
          numInstances_k = Sheets(dataSheet).Cells(k, ntuples + 3) 'columns are offset by 2
          Sheets(tempSheet).Cells(k, 1) = k - 3 'timestep
          Sheets(tempSheet).Cells(k, 2) = Sheets(dataSheet).Cells(k, 1) 'interval

          num_q_at_k = 0

          For z = 1 To ntuples 'traverse all the nTuples
            instances_q(z) = Sheets(dataSheet).Cells(k, z + 2)
            probabilities(z) = 0 'reinit array
            If instances_q(z) > 0 Then
              probabilities(z) = instances_q(z) / numInstances_k
            'ln = log(x) / log(exp(1))
              x_values(z) = Log(instances_q(z) + gamma) / Log(Exp(1)) 'Ln(instances_q(z) + gamma)
              y_values(z) = Log((-Log(1 - probabilities(z)) / Log(Exp(1)))) / Log(Exp(1)) 'Ln(-Ln(1 - probabilities(z)))

              num_q_at_k = num_q_at_k + 1

              'DEBUG********8
                Sheets(tempSheet).Cells(3, 10) = "actual probabilities"
```

```vba
            Sheets(tempSheet).Cells(k, z + 9) = probabilities(z)

        Else

        End If
      Next z
    If num_q_at_k > 0 Then
       coefficients = curveFit(num_q_at_k, x_values, y_values, 1)


      Sheets(tempSheet).Cells(k, 3) = coefficients(1) 'A
      Sheets(tempSheet).Cells(k, 4) = coefficients(2) 'B
      Sheets(tempSheet).Cells(k, 5) = coefficients(2) 'alpha = B
      If Not coefficients(2) = 0 Then
         Sheets(tempSheet).Cells(k, 6) = Exp(-coefficients(1) / coefficients(2)) 'beta
      Else
         Sheets(tempSheet).Cells(k, 6) = 0 'beta
      End If
    End If 'q at k > 0
  Next k


  'debug:::::::
  For k = 4 To numRows
    For z = 1 To ntuples

       instances_q(z) = Sheets(dataSheet).Cells(k, z + 2)
       If instances_q(z) > 0 Then
          'DEBUG*************
          '
          ' p(q_i) = 1-e^(q_i/beta)^alpha
          '
          If Sheets(tempSheet).Cells(k, 6) > 0 Then
            Dim p_q_i_calced As Double
            Dim beta As Double
            Dim alpha As Double
            beta = Sheets(tempSheet).Cells(k, 6).Value
            alpha = Sheets(tempSheet).Cells(k, 5).Value
            'Sheets(tempSheet).Cells(1, 1) = instances_q(z)
         '=(1-EXP(B3/$M$7)^$L$7)*-1
          '
            'p_q_i_calced = 1 - (1 / (Exp(instances_q(z) / beta) ^ alpha))
               Sheets(tempSheet).Cells(3, 40) = "calculated probabilities"
               Sheets(tempSheet).Cells(k, z + 39) = "=(1-EXP(-" & instances_q(z) & "/" & beta &
")^" & alpha & ")"
            End If
         '*****************
       End If
     Next z
   Next k
```

```
        End Sub
        '----------------------------------------------------------------------
        '  sub: fillMonthsRow
        '  Author: Matt Behnke
        '  Created: 12/11/01
        '  Description: fills in the months if they are missing.. Inserts a row
        '               works on lists.... must run 2 - 3 times to ensure all filled..
        '  inputs:  sheetName
        '  Outputs:
        '----------------------------------------------------------------------
        Sub fillMonthsRow(ByVal sheetName As String, ByVal startRow As Integer) ' ByVal sheetName
As String)

            'sheetName = ActiveSheet.Name

            Sheets(sheetName).Select
            numColumns = CountCols(sheetName, 1)
            numRows = CountRows(sheetName, 1)
            Dim theMonth As Integer

            'startRow = 4

            counter = 1
            monthCounter = "" & counter
            For i = startRow To numRows

               j = 1
               While found = False
                  testchar = Cells(i, 2).Characters(j, 1).Text
                  If testchar = "/" Then
                     found = True
                  Else
                  j = j + 1
                  End If
               Wend
               'month ends at j

               currentMonth = Cells(i, 2).Characters(1, j - 1).Text
               If currentMonth < 10 Then
                  restofDate = Cells(i, 2).Characters(2, 5).Text
               Else
                  restofDate = Cells(i, 2).Characters(3, 5).Text
               End If
               theMonth = currentMonth
               If theMonth > monthCounter Then
                  While theMonth > monthCounter
                     If i = startRow Then
                        'Range(Cells(i, 1), Cells(i, 1)).Select   DONT  ADD  TIMESTEPS  before  Starting
step!!

                        'Selection.EntireRow.Insert
```

407

```
            'Cells(i, 2) = "'" & monthCounter & restofDate
            'Cells(i, 3) = 0
            'Cells(i, 6) = 0
        Else
            Range(Cells(i, 1), Cells(i, 1)).Select
            Selection.EntireRow.Insert

        'copy previous row
            Rows(i - 1 & ":" & i - 1).Select
            Selection.Copy
            Rows(i & ":" & i).Select
            ActiveSheet.Paste
            Cells(i, 2) = "'" & monthCounter & restofDate
        End If

        Cells(i, 1) = i - startRow + 1
        counter = counter + 1
        If counter = 13 Then
           counter = 1
        End If
        monthCounter = "" & counter
        i = i + 1
        numRows = numRows + 1
      Wend
   End If

   Cells(i, 1) = i - startRow + 1

   counter = counter + 1
   If counter = 13 Then
      counter = 1
   End If
   monthCounter = "" & counter
   found = False
  Next i

End Sub ' fill months rows

Sub fillMonthsRowTrigger()

Call fillMonthsRow("A_Band_Stats", 14)
Call fillMonthsRow("B_Band_Stats", 14)
Call fillMonthsRow("C_Band_Stats", 14)
Call fillMonthsRow("D_Band_Stats", 14)
Call fillMonthsRow("A_Band_Stats", 14)
Call fillMonthsRow("B_Band_Stats", 14)
Call fillMonthsRow("C_Band_Stats", 14)
Call fillMonthsRow("D_Band_Stats", 14)

Call fillMonthsRow("World_Stats", 14)
```

```
Call fillMonthsRow("World_Stats", 14)

Call fillMonthsRow("Affiliation_Summary_A_Band", 4)
Call fillMonthsRow("Affiliation_Summary_B_Band", 4)
Call fillMonthsRow("Affiliation_Summary_C_Band", 4)
Call fillMonthsRow("Affiliation_Summary_D_Band", 4)
Call fillMonthsRow("Affiliation_Summary_A_Band", 4)
Call fillMonthsRow("Affiliation_Summary_B_Band", 4)
Call fillMonthsRow("Affiliation_Summary_C_Band", 4)
Call fillMonthsRow("Affiliation_Summary_D_Band", 4)

Call fillMonthsRow("Affiliation_Summary", 4)
Call fillMonthsRow("Affiliation_Summary", 4)
Call fillMonthsRow("Entropy Summary", 4)
Call fillMonthsRow("Entropy Summary", 4)
End Sub

'------------------------------------------------------------------------
' sub assignAN()
' Author: Matt Behnke
' Created: 10/19/01
' Description: Assigns month/year to each INSPEC Accession number
'              places the month/year on the title sheet, used to determine monthly values
'
'------------------------------------------------------------------------
Sub assignAN()

nRowsAN = CountRows("AN", 1)
nRowsTitle = CountRows("Title", 1)

Sheets("Title").Select
Sheets("Title").Cells(1, 3) = "PubDate"
Sheets("Title").Cells(1, 4) = "PubYear"

   For x = 2 To nRowsTitle
      compare = Sheets("Title").Cells(x, 1)
      For i = 2 To nRowsAN
         'find the accession number range that the AN from the title sheet
         'falls into
         timeInt = Sheets("AN").Cells(i, 1)
         startAN = Sheets("AN").Cells(i, 2)
         endAN = Sheets("AN").Cells(i, 3)
         If compare >= startAN And compare <= endAN Then
            Sheets("Title").Cells(x, 3) = timeInt   'when found put the month/year
            Sheets("Title").Cells(x, 4) = Year(timeInt)
         End If
      Next i
   Next x

End Sub
```

```
'----------------------------------------------------------------------
' sub putFirstAuthor()
' Author: Matt Behnke
' Created: 12/19/01
' Description: Fills in holes in the list of affiliations.
'     goes thru the list of authors.. if the accession number of the author is not found in affiliations,
'      place the first author name into the list of affiliations
'
'     can be used after to check to see if there are records without aff or author
'      After ran on author accession numbers run it on title accession numbers
'
'----------------------------------------------------------------------
Sub putFirstAuthor()

authors = "Authors (Cleaned)"
'authors = "Title"
affiliation = "Affiliation (Cleaned)"

nRowsAff = CountRows(affiliation, 1)
nRowsAuth = CountRows(authors, 1)
n = 1 'counter
found = False
lastAN = 0
For i = 2 To nRowsAuth
   authorNum = Sheets(authors).Cells(i, 1)
   authorName = Sheets(authors).Cells(i, 2)

   For j = 2 To nRowsAff
      affiliationNum = Sheets(affiliation).Cells(j, 1)
      If affiliationNum = authorNum Or lastAN = authorNum Then
         found = True
      End If
   Next j

   If found = False Then
      Sheets(affiliation).Cells(nRowsAff + n, 1) = authorNum
      Sheets(affiliation).Cells(nRowsAff + n, 2) = authorName
      n = n + 1
      lastAN = authorNum
   End If
   found = False
Next i

End Sub

'----------------------------------------------------------------------
' sub v_calc_v_psi_sheet()
' Author: Matt Behnke
' Created: 2/22/02
```

```vb
' Description: creates a sheet either by band or world that has the result of v and psi.
'           where v = (num records at an instance in step k)/(total num of records at step k)
'                ---------------------------------------------------------------------
'                (num of authors at an instance in step k)/(total num of authors at step k)
'
'           psi (tasks per timestep on avg) = v / timestep
'
' inputs: band - the source..
'
'---------------------------------------------------------------------
Sub v_calc_v_psi_sheet(ByVal band As String)

    'Dim authorMatrixTotals As Variant
    'Dim affiliationMatrixTotals As Variant
    Dim sum_v_array As Variant

    'N_i_k = the number of records produced by affiliation i at timestep k
    'N_Total_k = the number of records produced by all affiliations at timestep k
    'P_i_k = the number of authors who published in affiliation i at timestep k
    'P_Total_k = the number of authors who published in all affiliations at timestep k

    authorMatrixTotals = Array()
    affiliationMatrixTotals = Array()

    Sheets.Add After:=Worksheets(Worksheets.Count)

    Sheets(Worksheets.Count).Select

    ActiveSheet.Name = "v_calculation_" & band
    currentSheetName = ActiveSheet.Name

    'move the sheet so it is by related sheets
'    Sheets(currentSheetName).Move Before:=Sheets("" & band & "_Stats")

    If band = "World" Then
        affiliationMatrix = dataSheet
        authorMatrix = "Affiliation_authors"
    Else
        affiliationMatrix = "Affiliation_Cum_Dist_" & band
        authorMatrix = "Aff_Author_Cum_Dist_" & band
    End If

    numRowsInAffiliationMatrix = CountRows(affiliationMatrix, 1)
    numColsInAffiliationMatrix = CountCols(affiliationMatrix, 1)

    numRowsInAuthorMatrix = CountRows(authorMatrix, 1)
    numColsInAuthorMatrix = CountCols(authorMatrix, 1)

    'ReDim affiliationMatrixTotals(4 To numColsInAffiliationMatrix)
    'ReDim authorMatrixTotals(4 To numColsInAuthorMatrix)
```

```
ReDim sum_v_array(4 To numColsInAuthorMatrix)

'headers
Sheets(currentSheetName).Cells(1, 1) = " "
Sheets(currentSheetName).Cells(1, 2) = " "
Sheets(currentSheetName).Cells(3, 1) = "Time Step"
Sheets(currentSheetName).Cells(3, 2) = "interval"
Sheets(currentSheetName).Cells(3, 3) = "v"
Sheets(currentSheetName).Cells(3, 4) = "psi"

For i = 2 To numRowsInAuthorMatrix
  For j = 4 To numColsInAffiliationMatrix
    If i = 2 Then
      timeStep = j - 3
      interval = Sheets(authorMatrix).Cells(1, j)
      Sheets(currentSheetName).Cells(j, 1) = timeStep 'the timestep
      Sheets(currentSheetName).Cells(j, 2) = interval 'the timestep
    End If
    affiliationNameFromAuthorMatrix = Sheets(authorMatrix).Cells(i, 3)

    'find the row that contains the affiliation name from authors matrix in the affiliation matrix
    'sheet

    N_i_k_row = findStringInSheet(affiliationMatrix, affiliationNameFromAuthorMatrix, "C")

    'temp = Sheets(dataSheet).Cells(1, 1)
    'Sheets(dataSheet).Cells(1, 1) = N_i_k_range
    'N_i_k_row = Sheets(dataSheet).Cells(1, 1).Characters(4, 5).Text
    'Sheets(dataSheet).Cells(1, 1) = temp

    If j > 4 Then 'find the values at that instance... not cumulative
      P_i_k = Sheets(authorMatrix).Cells(i, j) - Sheets(authorMatrix).Cells(i, j - 1)
      N_i_k        =        Sheets(affiliationMatrix).Cells(N_i_k_row,       j)       -
Sheets(affiliationMatrix).Cells(N_i_k_row, j - 1)
        N_total = Sheets(affiliationMatrix).Cells(numRowsInAffiliationMatrix + 4, j) _
          - Sheets(affiliationMatrix).Cells(numRowsInAffiliationMatrix + 4, j - 1)
        P_total = Sheets(authorMatrix).Cells(numRowsInAuthorMatrix + 4, j) _
          - Sheets(authorMatrix).Cells(numRowsInAuthorMatrix + 4, j - 1)
    Else
      P_i_k = Sheets(authorMatrix).Cells(i, j)
      N_i_k = Sheets(affiliationMatrix).Cells(N_i_k_row, j)
      N_total = Sheets(affiliationMatrix).Cells(numRowsInAffiliationMatrix + 4, j)
      P_total = Sheets(authorMatrix).Cells(numRowsInAuthorMatrix + 4, j)
    End If
    'calculate v
    '    where v = (num records at an instance in step k)/(total num of records at step k)
    '         -----------------------------------------------------------------
    '         (num of authors at an instance in step k)/(total num of authors at step k)
    If P_i_k And N_total > 0 Then
      v = (((N_i_k / N_total) / P_i_k) / P_total)
```

```
            Else
                v = 0
            End If

            sum_v_array(j) = sum_v_array(j) + v
            'debug*****8
                Sheets(currentSheetName).Cells(i + 2, j + 4) = sum_v_array(j)
                Sheets(currentSheetName).Cells(1, 2) = j
            'debug
        Next j
        Sheets(currentSheetName).Cells(1, 1) = i
    Next i

    For j = 4 To numColsInAffiliationMatrix
    'output array of sum v.. make cumulative
        timeStep = j - 3


        If j > 4 Then 'cumulative
            Sheets(currentSheetName).Cells(j,      3)      =      sum_v_array(j)      +
Sheets(currentSheetName).Cells(j - 1, 3)
        Else
            Sheets(currentSheetName).Cells(j, 3) = sum_v_array(j)
        End If
        psi = Sheets(currentSheetName).Cells(j, 3) / timeStep

        Sheets(currentSheetName).Cells(j, 4) = psi
    Next j

End Sub 'calc_v_psi_sheet

'---------------------------------------------------------------------
' sub clearArray()
' Author: Matt Behnke
' Created: 2/22/02
' Description: clears the values stored in an array.
'
' inputs: lowerBound - lowerbound of the array
'         upperBound - upperbound of the array
'         arrayName - the array
'
'---------------------------------------------------------------------
Sub clearArray(ByVal lowerBound As Integer, ByVal upperBound As Integer, ByVal arrayName
As Variant)

    For i = lowerBound To upperBound
        arrayName(i) = ""
    Next i

End Sub
```

```
'----------------------------------------------------------------------
' function LinearInterpolation()
' Author: Matt Behnke
' Created: 2/26/02
' Description: linear interpolation used to calculate missing data:
'               [ X_i - X_low]
'   Y_i = y_low + [------------] * (Y_hi - Y_low)
'               [X_hi - X_low]
'
' inputs: Y_low - the closest "real" value to the left of the missing value
'         Y_high - the closest "real" value to the right of the missing value
'         X_low - the closest time step that has data to the left of the missing value
'         X_high - the closest time step that has data to the right of the missing value
'         X_i - the time step that has the missing data..
' output: returns a value, Y_i, for the missing time step.
'----------------------------------------------------------------------
    Function LinearInterpolation(ByVal Y_low As Double, ByVal Y_high As Double, ByVal X_low
As Integer, _
            ByVal X_high As Integer, ByVal X_i As Integer) As Double

        LinearInterpolation = Y_low + ((X_i - X_low) / (X_high - X_low)) * (Y_high - Y_low)

    End Function

'----------------------------------------------------------------------
' sub: FillInMissingData()
' Author: Matt Behnke
' Created: 2/26/02
' Description: Stores a list of data in an array, traverses the array to find
'           points where the data doesn't change. In our case where nothing was added
'           due to lack of information (small holes in the dataset).
'           When an element that doesn't change is found a linearInterpolation is performed
'           to determine what the value should be.
'            the value is changed and marked in red.
'
' inputs: dataSheet - the source of the data.
'         columnNumber - the column that contains the data
'         startRow - the row number where the data starts
'         endRow - the row number where the data ends
'----------------------------------------------------------------------
    Sub FillInMissingData(ByVal dataSheet As String, ByVal columnNumber As Integer, ByVal
startRow As Integer, _
            ByVal endRow As Integer)

        Dim dataArray As Variant
        dataArray = Array()

        numTimeSteps = endRow - startRow + 1
```

414

```
ReDim dataArray(1 To numTimeSteps)

' numTimesteps (k) = 5
' arrayindex (a) = 0 to 5
' startrow = 3
' endrow = 7
' k a   row
' 1 1  3+0=3
' 2 2  3+1=4
' 3 3  3+2=5
' 4 4  3+3=6
' 5 5  3+4=7

'populate the array with data
For i = 1 To numTimeSteps
    dataArray(i) = Sheets(dataSheet).Cells(startRow + i - 1, columnNumber)
Next i

'analyze the array:
For i = 1 To numTimeSteps
  If Not i = numTimeSteps Then
     currentValue = dataArray(i)
     nextValue = dataArray(i + 1)
     If currentValue = nextValue Then
        lowestDifferentValue = dataArray(i)              'Y_low
        lowestDifferentValuePosition = i                 'X_low
        For j = i + 2 To numTimeSteps 'scan the array to find the next higher value
           nextHigherValue = dataArray(j)                'Y_high
           If Not nextHigherValue = currentValue Then
              nextHigherValuePosition = j                'X_high
              Exit For 'j
           End If
        Next j

        For k = lowestDifferentValuePosition + 1 To nextHigherValuePosition - 1
        'now the next lower and higher values are known along with their positions,call
linearInterpolation
           'k = y_i
             dataArray(k) = LinearInterpolation(lowestDifferentValue, nextHigherValue, _
                          lowestDifferentValuePosition, nextHigherValuePosition, k)
        Next k
     End If 'currentvalue = nextValue
  End If 'not equal to numTimesteps
Next i

'output the array
For i = 1 To numTimeSteps
    Sheets(dataSheet).Cells(startRow + i - 1, columnNumber) = dataArray(i)
Next i
```

End Sub 'fill in missing data

'TEST FILL IN MISSING DATA
Sub testFillInMissing()

'tests the linear interpolation function..can also be used as an interface to the function...

dataSheet = InputBox("enter the name of the source sheet")
columnNumber = InputBox("enter column number")
rowStart = InputBox("enter the first row of data")
rowEnd = InputBox("enter the last row of data")

   Call FillInMissingData(dataSheet, columnNumber, rowStart, rowEnd)

End Sub

'---
' CurveFit
' Author: Erchuang (Al) Wang (original), Matt Behnke - converted to VB
' Converted: 2/27/02
' Description:
'   This program will fit a curve up to 10th degree polynomial
'   in the form of Y = a0 + a1*x + a2*x^2 + ... + a(n)*x^(n)
'   where n is the degree of the polynomial and 1>=n=<30
'   Reads in two lists of numbers, X & Y-values and performs the fit
'
' inputs: dataSheet - sheet with the source values
'   numValues - the number of values in the array
'   x - array of the x values
'   y - array of the y-values
'   degree - the degree of the polynomial
' Outputs: CoefficientArray - the coefficients of the equation a0, a1, ... a(n)
'---
Function curveFit(ByVal numValues As Integer, _
            ByVal x As Variant, ByVal y As Variant, ByVal degree As Integer) As Variant

   'numValues = endRow - startRow + 1

   'variables
   Dim coefficientArray(64) As Variant   'stores the results, max of 64 coeff..
   'Dim x As Variant   'a one dimension array for x values
   'Dim y As Variant   'a one dimension array for y values
   Dim cn(64) As Variant   '???????????????????????????

   Dim ar(64, 64) As Variant   'a two dimension array
   Dim an(64, 64) As Variant   'answer array

   Dim sum As Double

416

```
Dim t As Double
Dim d As Double
Dim b As Double

Dim j As Integer                              'for loop counter
Dim i As Integer                              'for loop counter
Dim m As Integer
Dim n As Integer                               '=numValues, number of data points
Dim ii As Integer                             'for loop counter
Dim k As Integer                               'for loop counter
Dim nn As Integer
Dim nd As Integer                               '=degree, degree of poly


n = numValues
nd = degree
m = nd + 1
nn = m + 1

'generate normal equation A and vector B of Ax=B
For ii = 1 To n
   For j = 1 To m
      If j = 1 And x(ii) = 0# Then
         ar(ii, j) = 1#
      Else
         ar(ii, j) = x(ii) ^ (j - 1)
      End If
   Next j
Next ii

For k = 1 To m
   For ii = 1 To m
      sum = 0#
      For j = 1 To n
         sum = sum + ar(j, k) * ar(j, ii)
      Next j
      an(k, ii) = sum
   Next ii
Next k

For ii = 1 To m
   sum = 0#
   For j = 1 To n
      sum = sum + y(j) * ar(j, ii)
   Next j
   cn(ii) = sum
Next ii

'solve x vector of Ax=B where A=A'

For i = 1 To m
```

```
      an(i, nn) = cn(i)
   Next i

   For i = 1 To m
      k = i
      b = Abs(an(i, i))
      If b = 0# Then
         For j = i To m
            If b < Abs(an(j, ii)) Then
               b = Abs(an(j, i))
               k = j
            End If
         Next j

         For j = 1 To nn
            t = an(i, j)
            an(i, j) = an(k, j)
            an(k, j) = t
         Next j

      Else
         d = an(i, i)
      End If

      For j = 1 To nn
         an(i, j) = an(i, j) / d
      Next j

      For j = 1 To m
         b = an(j, i)

         For k = 1 To nn
            If Not j = i Then
               an(j, k) = an(j, k) - an(i, k) * b
            End If
         Next k
      Next j
   Next i

   'put answers into the coefficient array

   For ii = 1 To m
      coefficientArray(ii) = an(ii, nn)
   Next ii

   curveFit = coefficientArray

End Function 'curvefit

Sub testFind()
```

```
'WORKS'...
"Phys. Dept., Kakatiya Univ., Warangal, India
With ActiveSheet.Range("A:A")
   Set C = .Find("3/1979", LookIn:=xlValues)
   If Not C Is Nothing Then
      firstAddress = C.Address
      MsgBox (firstAddress)
   End If
End With
'MsgBox (Search("Phys. Dept., Kakatiya Univ., Warangal, India"))

End Sub
```

```
'-------------------------------------------------------------------------
'  Macro: Cumulative
'  Author: Matt Behnke
'  Created: 9/10/01
'  Description: Computes the cumulative entopy for each term at each time
'               interval.
'               Creates summary sheets and graphs based on the computation.
'-------------------------------------------------------------------------
Sub Cumulative()

'declare constants:
  dataSheet = "Sheet1"
  copyTo = "Sheet2"
  summarySheet = "Sheet3"
  sliceStart = 4 'first column of timeslices
  termStart = 4 'first row of that contains a term

'fill empty cells in on the datasheet so count rows function works properly
  Sheets("" & dataSheet).Select
  Cells(2, 1) = " "
  Cells(1, 1) = " "
  Cells(1, 2) = " "

  Call copyTerms(dataSheet, copyTo)
  Call addSums(dataSheet, termStart, sliceStart, copyTo)
  Call fillSheets(dataSheet, copyTo, sliceStart, termStart)
  Call removeFormulas(copyTo, dataSheet, termStart, sliceStart)

  Call createGraph(copyTo, dataSheet, termStart, sliceStart, "Entropy Power Trend", xlPower)
  Call createSummary(dataSheet, copyTo, termStart, sliceStart)
  Call entropyLambda(summarySheet, copyTo, termStart, sliceStart)

  Sheets("" & dataSheet).Name = "Data"                                    '(R12.1)
  Sheets("" & copyTo).Name = "Cumlative_Entropy"                          '(R12.2)
  Sheets("" & summarySheet).Name = "Summary"                             '(R5.12)

End Sub


'-------------------------------------------------------------------------
'  Subroutine: copyTerms
'  Author: Matt Behnke
'  Created: 9/10/01
'  Description:  1) Copies the terms from the data sheet to another sheet where
'                the entropy formula will be applied to the data.           (R2.1)
'  inputs:   dataSheet - name of the datasheet
'            copyTo - name of the sheet with the copied terms
'  Outputs: none
'-------------------------------------------------------------------------
Sub copyTerms(ByVal dataSheet As String, ByVal copyTo As String)
  termEnd = CountRows(dataSheet, 1)
  sliceEnd = CountCols(dataSheet, 3)

  Worksheets("Sheet1").Range("A1:"          &          col(sliceEnd)          &          termEnd).Copy
Destination:=Worksheets("Sheet2").Range("A1")
```

```
          Sheets("" & copyTo).StandardWidth = 9                                    '(R12.3)

       End Sub 'copyTerms


       '----------------------------------------------------------------------------
       ' Subroutine: addSums
       ' Author: Matt Behnke
       ' Created: 9/10/01
       ' Description:  1) Puts the sum of all the term's instances for each time
       '                  interval one row below the data in that interval          (R2.2)
       '               2) Puts the cumulative sum of term instances from previous
       '                  time intervals one row below the sum of term instances.   (R2.3)
       ' inputs:   dataSheet - name of the datasheet
       '           copyTo - name of the sheet with the copied terms
       '           sliceStart - start of time slice columns
       '           termStart - start of the terms (rows)
       ' Outputs: none
       '----------------------------------------------------------------------------
       Sub addSums(ByVal dataSheet As String, ByVal sliceStart As Integer, ByVal termStart As
Integer, ByVal copyTo As String)

          sliceEnd = CountCols(dataSheet, 3)
          x = CountRows(dataSheet, 1) 'last row of the terms

          Sheets("" & dataSheet).Cells(x + 1, 3) = "Sum"
          Sheets("" & dataSheet).Cells(x + 1, 3) = "Sum to date"

          For i = sliceStart To sliceEnd
          'for each column in time slice range put the formula that calcs local sum
            Sheets("" & dataSheet).Cells(x + 1, i).Formula = "=SUM(" & col(i) & termStart & ":" & col(i)
& x & ")"
            'place formula on copied sheet also.. where entropy sums will be          (R2.5)
            Sheets("" & copyTo).Cells(x + 1, i).Formula = "=SUM(" & col(i) & termStart & ":" & col(i) &
x & ")"

            'cumulative number of instances per slice:
            If i = sliceStart Then
               Sheets("" & dataSheet).Cells(x + 2, i).Formula = "=" & col(i) & x + 1
            Else
               Sheets("" & dataSheet).Cells(x + 2, i).Formula = "=" & col(i) & x + 1 & "+" & col(i - 1) & x
+ 2
            End If
          Next i

       'format the datasheet for print (R12.4)
         Sheets("" & dataSheet).Select
         Call formatSheetForPrint

       End Sub 'addSums
       '----------------------------------------------------------------------------
       ' Subroutine: fillSheets
       ' Author: Matt Behnke
       ' Created: 9/11/01
       ' Description:  1) Places the formula used to calculate the cumulative entropy
       '                  in each row of terms in the first time slice.
```

```
'              2) Calls copy formula to copy the formula from the first time slice
'                  to all of them (R2.4)
' inputs:   dataSheet - name of the datasheet
'           copyTo - name of the sheet with the copied terms
'           sliceStart - start of time slice columns
'           termStart - start of the terms (rows)
' Outputs: none
'-------------------------------------------------------------------------

Sub fillSheets(ByVal dataSheet As String, ByVal copyTo As String, ByVal sliceStart As Integer, ByVal termStart As Integer)

    termEnd = CountRows(dataSheet, 1)
    sliceEnd = CountCols(dataSheet, 3)

    i = sliceStart

    For x = termStart To termEnd
      Sheets("" & copyTo).Cells(x, i).Formula = "=If(SUM(" & dataSheet & "!$" & col(sliceStart) & x & _
        ":" & dataSheet & "!" & col(i) & x & ")=0,0, -SUM(" & dataSheet & "!$" & col(sliceStart) & x & _
        ":" & dataSheet & "!" & col(i) & x & ")/" & dataSheet & "!" & col(i) & termEnd + 2 & "*" _
        & "LOG(SUM(" & dataSheet & "!$" & col(sliceStart) & x & _
        ":" & dataSheet & "!" & col(i) & x & ")/" & dataSheet & "!" & col(i) & termEnd + 2 & ",2))"
    Next x

    'format the entropy data sheet (R12.4)
      Sheets("" & copyTo).Select
      Call formatSheetForPrint
      With Worksheets("" & copyTo).Columns("C")                          '(R12.7)
          .ColumnWidth = 43
      End With
      With Worksheets("" & dataSheet).Columns("C")
          .ColumnWidth = 43
      End With
    'call copy formulas subroutine to finish calculation.
      Call copyFormulas(copyTo, dataSheet, termStart, sliceStart)        '(R2.4)
    End Sub 'fillSheets

    '-------------------------------------------------------------------------
    ' Subroutine: copyFormulas
    ' Author: Matt Behnke
    ' Created: 9/11/01
    ' Description:  1) copies the formulas from the first time interval to the rest of the
    '                 intervals
    '
    ' inputs:   copyTo - name of the sheet with the copied terms
    '           sliceStart - start of time slice columns
    '           termStart - start of the terms (rows)
    ' Outputs: none
    '-------------------------------------------------------------------------
    Sub copyFormulas(ByVal copyTo As String, ByVal termStart As Integer, ByVal sliceStart As Integer)
```

422

```vbnet
    termEnd = CountRows(copyTo, 1)
    sliceEnd = CountCols(copyTo, 3)
'select the column of the first time slice where the entropy formula has been applied.
    Sheets("" & copyTo).Select
    Range("" & col(sliceStart) & termStart & ":" & col(sliceStart) & termEnd).Select
      Selection.Copy

'copy the formula from the first time slice's column to every other time slices' column
    For i = sliceStart + 1 To sliceEnd
      Range("" & col(i) & termStart).Select
        ActiveSheet.Paste
    Next i

End Sub

'-------------------------------------------------------------------------
' Subroutine: removeFormulas
' Author: Matt Behnke
' Created: 9/14/01
' Description:  1) removes the formulas from the copiedTo sheet (where cumulative entropy
'                 is) .. This gives faster worksheet loading time because the cells
'                 don't need to be calulated everytime the worksheet is loaded.
' inputs:   copyTo - name of the sheet with the copied terms
'           sliceStart - start of time slice columns
'           termStart - start of the terms (rows)
' Outputs: none
'-------------------------------------------------------------------------
Sub removeFormulas(ByVal copyTo As String, ByVal termStart As Integer, ByVal sliceStart As
Integer)

    termEnd = CountRows(copyTo, 1)
    sliceEnd = CountCols(copyTo, 3)
'copy the sheet and paste special (values only)
    Sheets("" & copyTo).Select
    Range("" & col(sliceStart) & termStart & ":" & col(sliceEnd) & termEnd).Select
    Selection.Copy
    Range("" & col(sliceStart) & termStart).Select
    Selection.PasteSpecial Paste:=xlValues, Operation:=xlNone, SkipBlanks:= _
        False, Transpose:=False
End Sub

'-------------------------------------------------------------------------
' Subroutine: createGraph
' Author: Matt Behnke
' Created: 9/12/01
' Revised: 9/14, 9/17
' Description: 1) Creates a chart and names it according the the name in the input.
'             2) creates a trend-line on the source data. added 9/17
'             3) formats titles, data series markers, trend-line, chart area (9/10, 14, 17)
' inputs:   sourceSheet - name of the sheet where cumulative entropy has been calculated
'           dataSheet - original data sheet (co-occurance matrix from Tech OASIS)
'           termStart - start of the terms (rows)
'           sliceStart - start of time slice columns
'           chartName - name of the chart
'           trendType - type of trendline to add
```

```
' Outputs: none
'---------------------------------------------------------------------------
Sub createGraph(ByVal sourceSheet As String, ByVal dataSheet As String, ByVal termStart As
Integer, ByVal sliceStart As Integer, ByVal chartName As String, ByVal trendType As String)

        termEnd = CountRows(dataSheet, 1)
        sliceEnd = CountCols(dataSheet, 3)
        projectYears = (sliceEnd - 4) / 2 'calculate number of units to project trend-line      (R3.5)

        Charts.Add
        ActiveChart.ChartType = xlXYScatterSmooth                                               '(R3.1)
        ActiveChart.SetSourceData Source:=Sheets("" & sourceSheet).Range("" & col(sliceStart) &
termEnd + 1 & ":" _
        & col(sliceEnd) & termEnd + 1), PlotBy:=xlRows                                          '(R3.2)
        ActiveChart.Location Where:=xlLocationAsNewSheet
        With ActiveChart
          .HasLegend = True
          .HasTitle = True
          .ChartTitle.Characters.Text = "Cumulative Entropy vs. Year"                           '(R4.1)
          .Axes(xlCategory, xlPrimary).HasTitle = True
          .Axes(xlCategory, xlPrimary).AxisTitle.Characters.Text = "k (Years)"                  '(R4.3)
          .Axes(xlValue, xlPrimary).HasTitle = True
          .Axes(xlValue, xlPrimary).AxisTitle.Characters.Text = "Entropy Sk (Bits)"            '(R4.2)
        End With

        'increase chart area and move legend                                                   (R4.7,
R4.8)
          ActiveChart.PlotArea.Select
          Selection.Width = 598
          Selection.Height = 395
          ActiveChart.Legend.Select
          Selection.Left = 326
          Selection.Top = 207

        'change line style and marker points style                                             (R4.5,
R4.6)
          ActiveChart.SeriesCollection(1).Name = "Cumulative Entropy"
          ActiveChart.SeriesCollection(1).Select
          With Selection.Border
            .Weight = xlThin
            .LineStyle = xlNone
          End With
          With Selection
            .MarkerBackgroundColorIndex = 44
            .MarkerForegroundColorIndex = 45
            .MarkerStyle = xlTriangle
            .Smooth = True
            .MarkerSize = 6
            .Shadow = True
          End With

         currentName = ActiveChart.Name
        'add trendline                                                                         (R3.3)
          ActiveWorkbook.Charts("" & currentName).SeriesCollection(1).Trendlines.Add
```

424

```
    Sheets("" & dataSheet).Select
    With Charts("" & currentName).SeriesCollection(1).Trendlines(1)
      .Type = trendType                                              '(R3.4)
      .Forward = projectYears                                        '(R3.6)
      .DisplayEquation = True                                        '(R4.9)
      If trendType = xlPower Then
        Worksheets("sheet3").Cells(1, 4).Value = .DataLabel.Text
      End If
      .DisplayRSquared = True
    End With

'move trendline                                   '(R4.10)
    ActiveChart.SeriesCollection(1).Trendlines(1).DataLabel.Select
      Selection.Left = 494
      Selection.Top = 198

'increase legend size
    ActiveChart.Legend.Select
      Selection.Width = 201

'remove border and color fill on plot area
    ActiveChart.PlotArea.Select
      With Selection.Border
        .Weight = xlHairline
        .LineStyle = xlNone
      End With
      Selection.Interior.ColorIndex = xlNone

'remove border on legend
    ActiveChart.Legend.Select
      With Selection.Border
        .Weight = xlHairline
        .LineStyle = xlNone
      End With

    Call formatChartForPrint(currentName)                              '(R4.11,
R4.12)

    Sheets("" & currentName).Select
    Sheets("" & currentName).Name = chartName                          '(R3.1)

End Sub 'CreateGraph
'--------------------------------------------------------------------------
'  Subroutine: createSummary
'  Author: Matt Behnke
'  Created: 9/17/01
'  Description: 1) Creates a summary sheet showing cumulative entropy in each Time interval.
'               2) calculates predicted values of cumulative entropy based on the trendline equation
'                  from the cumulative entropy graph.
'               3) calculates the percent error btw actual and predicted values.
'  inputs:   dataSheet - original data sheet name (co-occurance matrix from Tech OASIS)
'            copyTo - sheetname of sheet containing calculated entropy values
'            termStart - start of the terms (rows)
'            sliceStart - start of time slice columns
'  Outputs: none
```

```vba
'------------------------------------------------------------------------
Sub createSummary(ByVal dataSheet As String, ByVal copyTo As String, ByVal termStart As Integer, ByVal sliceStart As Integer)

    termEnd = CountRows(dataSheet, 1)
    sliceEnd = CountCols(dataSheet, 3)

    Sheets("Sheet3").StandardWidth = 16                                          '(R5.10)
    Sheets("Sheet3").Cells(1, 1) = "Time T"                                      '(R5.1)
    Sheets("Sheet3").Cells(1, 2) = "Slice"
    Sheets("Sheet3").Cells(1, 3) = "Cum Entropy (Actual)"
    Sheets("Sheet3").Cells(1, 5) = "Predicted: " & Chr(10) & "5 years of data"
    Sheets("Sheet3").Cells(1, 6) = "Predicted: " & Chr(10) & "10 years of data"
    Sheets("Sheet3").Rows("1:1").RowHeight = 38.25                               '(R5.11)
    Count = 1

'get the error formula from the power trendline

    firstPart = Sheets("Sheet3").Cells(1, 4).Characters(5, 5).Text              '(R5.5)
    secondPart = Sheets("Sheet3").Cells(1, 4).Characters(12, 5).Text            '(R5.6)

    For i = sliceStart To sliceEnd
        sliceName = Sheets("" & dataSheet).Cells(termStart - 1, i)
        Sheets("Sheet3").Cells(i - 2, 1) = Count                                '(R5.2)
        Sheets("Sheet3").Cells(i - 2, 2) = sliceName                            '(R5.3)
        Sheets("Sheet3").Cells(i - 2, 3) = Sheets("" & copyTo).Cells(termEnd + 1, i)   '(R5.4)
        Entropy = Sheets("Sheet3").Cells(i - 2, 3)
        Sheets("Sheet3").Cells(i - 2, 4).Formula = "=((" & firstPart & "*A" & i - 2 & "^" _
        & secondPart & ")-" & Entropy & ")/" & Entropy                         '(R5.7)
        Count = Count + 1
'project 5 years                                                                '(R5.8)
        If Count <= 6 Then
            Sheets("Sheet3").Cells(i - 2, 5) = Entropy
        Else
            Sheets("Sheet3").Cells(i - 2, 5) = "=" & firstPart & "*" & "A" & i - 2 & "^" & secondPart
        End If
'project ten years                                                              '(R5.9)
        If Count <= 11 Then
            Sheets("Sheet3").Cells(i - 2, 6) = Entropy
        Else
            Sheets("Sheet3").Cells(i - 2, 6) = "=" & firstPart & "*" & "A" & i - 2 & "^" & secondPart
        End If

    Next i

    Sheets("Sheet3").Select
    Range("D:D").Select
    Selection.NumberFormat = "0.00%"

    Call formatSheetForPrint                                                    '(R12.4)

End Sub 'createSummary
'------------------------------------------------------------------------
' Subroutine: entropyLambda
' Author: Matt Behnke
```

```
' Created: 9/19/01
' Revised: 9/24 - added map k, k+1 stuff
' Description: 1) Creates a sheet called "entropy lambda" where lambda and the Lyaponuv
'               number is calculated based on formulas given in the requirements
' inputs:  summarySheet - name of the summary sheet
'          copyTo - sheetname of sheet containing calculated entropy values
'          termStart - start of the terms (rows)
'          sliceStart - start of time slice columns
' Outputs: none
'--------------------------------------------------------------------------
Sub entropyLambda(ByVal summarySheet As String, ByVal copyTo As String, ByVal termStart
As Integer, ByVal sliceStart As Integer)

    termEnd = CountRows(copyTo, 1)
    sliceEnd = CountCols(copyTo, 3)

'create new sheet                                               '(R6.1)
  Sheets.Add
  currentName = ActiveSheet.Name
  Sheets("" & currentName).Name = ("EntropyLambda")
  currentName = ActiveSheet.Name

'set height of header row and standard column width
  Rows("1:1").RowHeight = 52                                    '(R12.6)
  ActiveSheet.StandardWidth = 12                                '(R12.5)

'copy first four columns of summary sheet                       '(R6.2)
  Sheets("" & summarySheet).Select
  Range("A1:D" & sliceEnd).Select
    Selection.Copy
    Range("A1").Select
  Sheets("" & currentName).Select
    ActiveSheet.Paste

  numRows = CountRows(currentName, 1)

'copy object (equation objects that display below the data) from macro:
    workbookName = ActiveWorkbook.Name
    ActiveWindow.SmallScroll Down:=-6
    Range("G4").Select
    ActiveWindow.SmallScroll Down:=-9
    Windows("CumEntropyMacro2.xls").Activate
    ActiveSheet.Shapes("Group 5").Select
    Selection.Copy
    Windows("" & workbookName).Activate
    Range("A" & sliceEnd + 3).Select
    ActiveSheet.Paste
    Range("G31").Select

'headers                                                        '(R6.3)
  ActiveSheet.Cells(1, 5) = "Cum_K+1"
  ActiveSheet.Cells(1, 7) = "du_(t-c)"
  ActiveSheet.Cells(1, 8) = "du_(t)"
  ActiveSheet.Cells(1, 9) = "du_(t-2)"
  ActiveSheet.Cells(1, 10) = "du_(t-5)"
```

427

```
     ActiveSheet.Cells(1, 11) = "du_(t-15)"
     ActiveSheet.Cells(1, 12) = "du_(t-20)"
     ActiveSheet.Cells(1, 13) = "C_y_10%"
     ActiveSheet.Cells(1, 14) = "Lambda_B10%_y"
     ActiveSheet.Cells(1, 15) = "B10%_y"
     ActiveSheet.Cells(1, 16) = "C_y_20%"
     ActiveSheet.Cells(1, 17) = "Lambda_B20%"
     ActiveSheet.Cells(1, 18) = "B20%_y"
     ActiveSheet.Cells(1, 19) = "C_y_50%"
     ActiveSheet.Cells(1, 20) = "Lambda_B50%"
     ActiveSheet.Cells(1, 21) = "B50%_y"
     ActiveSheet.Cells(1, 22) = "C_y_100"
     ActiveSheet.Cells(1, 23) = "Lambda_B100"
     ActiveSheet.Cells(1, 24) = "B100_y"

'fill in column 5: Cum_K+1                                                    '(R6.10)
   For i = 2 To numRows
     ActiveSheet.Cells(i, 5) = "=C" & i + 1
   Next i

'create the map of k, k+1 to get the trendline equation for                   '(R6.11)
'calculating the lyanponuv exponent.
   Call createMapEntropyKK_1(numRows)
   Sheets("" & currentName).Select

'get formula of trendline from entropy power trend graph
   firstPart = ActiveSheet.Cells(1, 4).Characters(5, 5).Text                  '(R6.4)
   secondPart = ActiveSheet.Cells(1, 4).Characters(12, 5).Text                '(R6.5)

'get formula of trendline from entropy map k, k+1
   firstPartMap = ActiveSheet.Cells(1, 6).Characters(5, 5).Text               '(R6.12)
   secondPartMap = ActiveSheet.Cells(1, 6).Characters(12, 5).Text             '(R6.13)

'rename column 4 header.. in summary and entropy lambda sheets                '(R6.3.4)
   Sheets("Sheet3").Cells(1, 4) = "Error (Act vs. Pred)" & Chr(10) & Sheets("Sheet3").Cells(1, 4)
   ActiveSheet.Cells(1, 4) = "Error (Act vs. Pred)" & Chr(10) & ActiveSheet.Cells(1, 4)

'rename column 6 header                                                       '(R6.3.6)
   ActiveSheet.Cells(1, 6) = "Lyaponuv  Exp J'(k,k+1) = " & Chr(10) & secondPartMap & "*" & _
     firstPartMap & " k^(" & secondPartMap & "-1)"

'change column width of column 4, 6 (Error column)                           '(R12.8)
With Worksheets("" & currentName).Columns("D")
     .ColumnWidth = 16
End With
With Worksheets("" & currentName).Columns("F")
     .ColumnWidth = 16
End With

'place m * b calculation of power trend in column 6 at the end of the data    '(R6.6)
   ActiveSheet.Cells(sliceEnd + 2, 6) = "m*b"
   ActiveSheet.Cells(sliceEnd + 3, 6) = "" & firstPart & "*" & secondPart
   ActiveSheet.Cells(sliceEnd + 4, 6) = "=" & firstPart & "*" & secondPart

'place lyaponuv stuff below m*b stuff                                         '(R6.14)
```

```vba
    ActiveSheet.Cells(sliceEnd + 6, 6) = "J '(k,k+1)= " & secondPartMap & "*" & _
      firstPartMap & " k^(" & secondPartMap & "-1)"
    ActiveSheet.Cells(sliceEnd + 7, 6) = "" & firstPartMap & "*" & secondPartMap
    ActiveSheet.Cells(sliceEnd + 8, 6) = "=" & firstPartMap & "*" & secondPartMap
    ActiveSheet.Cells(sliceEnd + 9, 6) = "=" & secondPartMap & "-1"
    ActiveSheet.Cells(sliceEnd + 8, 7) = "J' coeff"
    ActiveSheet.Cells(sliceEnd + 9, 7) = "J' exponent"

  'fill in lyaponuv data in column 6                                      '(R6.15)
   jcoeff = ActiveSheet.Cells(sliceEnd + 8, 6)
   jexp = ActiveSheet.Cells(sliceEnd + 9, 6)
   For i = 2 To numRows
     ActiveSheet.Cells(i, 6) = "=" & jcoeff & "*C" & i & "^" & jexp
   Next i

  'place du equations in column 7 below data                             '(R6.7)
    ActiveSheet.Cells(sliceEnd + 2, 7) = "du = (" & firstPart & "*" & secondPart & ")*T^(" & _
       secondPart & "-1)"

    ActiveSheet.Cells(sliceEnd + 3, 7) = "du_t-c = (" & firstPart & "*" & secondPart & ")*T_t-c^(" & _
       secondPart & "-1)"

  'fill in the formulas for derivatives                                  '(R6.8)
     For i = sliceStart - 1 To sliceEnd - 2
       ActiveSheet.Cells(i, 7) = "=" & "$" & col(6) & sliceEnd + 4 & "*$A" & i - 1 & "^(" & _
secondPart & "-1)"
       ActiveSheet.Cells(i, 8) = "=" & "$" & col(6) & sliceEnd + 4 & "*$A" & i & "^(" & secondPart & "-1)"

       If i >= 4 Then
          ActiveSheet.Cells(i, 9) = "=" & "$" & col(6) & sliceEnd + 4 & "*$A" & i - 2 & "^(" & _
secondPart & "-1)"
       End If

       If i >= 7 Then
          ActiveSheet.Cells(i, 10) = "=" & "$" & col(6) & sliceEnd + 4 & "*$A" & i - 5 & "^(" & _
secondPart & "-1)"
       End If

       If i >= 17 Then
          ActiveSheet.Cells(i, 11) = "=" & "$" & col(6) & sliceEnd + 4 & "*$A" & i - 15 & "^(" & _
secondPart & "-1)"
       End If

       If i >= 22 Then
          ActiveSheet.Cells(i, 12) = "=" & "$" & col(6) & sliceEnd + 4 & "*$A" & i - 20 & "^(" & _
secondPart & "-1)"
       End If

  'fill in the values in columns 13-24                                   '(R6.9)
     '10%
     ActiveSheet.Cells(i, 13) = "=" & "$" & col(3) & i & "-" & col(14) & i
     ActiveSheet.Cells(i, 14) = "=(" & "$" & col(15) & i & "*" & col(7) & i & "/((" & col(15) & i _
       & "*" & col(7) & i & ")+" & col(8) & i & "))^(1/3)"
```
429

```
    ActiveSheet.Cells(i, 15) = 0.1

    '20%
    ActiveSheet.Cells(i, 16) = "=" & "$" & col(3) & i & "-" & col(17) & i
    ActiveSheet.Cells(i, 17) = "=(" & "$" & col(18) & i & "*" & col(7) & i & "/((" & col(18) & i _
    & "*" & col(7) & i & ")+" & col(8) & i & "))^(1/3)"
    ActiveSheet.Cells(i, 18) = 0.2

    '50%
    ActiveSheet.Cells(i, 19) = "=" & "$" & col(3) & i & "-" & col(20) & i
    ActiveSheet.Cells(i, 20) = "=(" & "$" & col(21) & i & "*" & col(7) & i & "/((" & col(21) & i _
    & "*" & col(7) & i & ")+" & col(8) & i & "))^(1/3)"
    ActiveSheet.Cells(i, 21) = 0.5

    '75%
    ActiveSheet.Cells(i, 22) = "=" & "$" & col(3) & i & "-" & col(23) & i
    ActiveSheet.Cells(i, 23) = "=(" & "$" & col(24) & i & "*" & col(7) & i & "/((" & col(24) & i _
    & "*" & col(7) & i & ")+" & col(8) & i & "))^(1/3)"
    ActiveSheet.Cells(i, 24) = 0.75

  Next i

  Call formatSheetForPrint                                              '(R12.4)
  Call createLambdaChart

  End Sub 'entropyLambda


  '-------------------------------------------------------------------------
  ' Subroutine: createMapEntropyKK_1
  ' Author: Matt Behnke
  ' Created: 9/24/01
  ' Description: Called in entropyLambda, it creates the chart of entropy K and K+1
  '              gets the equation from the power trendline y=mx^b and puts it on the entropy
  '              lambda sheet so it can be used.
  ' inputs: number of rows of data on the entropy lambda sheet
  ' Outputs: none
  '-------------------------------------------------------------------------
  Sub createMapEntropyKK_1(ByVal numRows As Integer)

    Charts.Add                                                          '(R9.1)
    ActiveChart.ChartType = xlXYScatterSmooth
    ActiveChart.SetSourceData Source:=Sheets("EntropyLambda").Range("C2:C" & numRows - 1),
  PlotBy:=xlColumns
    ActiveChart.Location Where:=xlLocationAsNewSheet

  'change name and add axis labels
    With ActiveChart
     .HasLegend = True                                                 '(R10.4)
     .HasTitle = True
     .ChartTitle.Characters.Text = "Entropy Finite Difference Mapping Sk+1=f(Sk)"   '(R10.1)
     .Axes(xlCategory, xlPrimary).HasTitle = True
     .Axes(xlCategory, xlPrimary).AxisTitle.Characters.Text = "Entropy Sk"          '(R10.2)
     .Axes(xlValue, xlPrimary).HasTitle = True
     .Axes(xlValue, xlPrimary).AxisTitle.Characters.Text = "Entropy Sk+1"           '(R10.3)
    End With
```

```
'increase size of chart and move legend                                          '(R10.5,
R10.6)
       ActiveChart.PlotArea.Select
         Selection.Width = 598
         Selection.Height = 395
         ActiveChart.Legend.Select
         Selection.Left = 380
         Selection.Top = 300

      'increase legend size
        ActiveChart.Legend.Select
         Selection.Width = 201

      'remove border and color fill on plot area
        ActiveChart.PlotArea.Select
          With Selection.Border
            .Weight = xlHairline
            .LineStyle = xlNone
          End With
          Selection.Interior.ColorIndex = xlNone

      'remove border on legend
        ActiveChart.Legend.Select
          With Selection.Border
            .Weight = xlHairline
            .LineStyle = xlNone
          End With

      'change line style and marker points style                                 '(R10.9,
R10.10)
         ActiveChart.SeriesCollection(1).Name = "Entropy Map S_k+1, S_k"          '(R9.2)
         ActiveChart.SeriesCollection(1).Select
           With Selection.Border
             .Weight = xlThin
             .LineStyle = xlNone
           End With
           With Selection
             .MarkerBackgroundColorIndex = 44
             .MarkerForegroundColorIndex = 45
             .MarkerStyle = xlTriangle
             .Smooth = True
             .MarkerSize = 6
             .Shadow = True
           End With

      'format axis and set the correct source data                               '(R9.3)
        ActiveChart.SeriesCollection(1).Select
        ActiveChart.SeriesCollection(1).XValues = "=EntropyLambda!R2C3:R" & numRows - 1 &
"C3"
         ActiveChart.SeriesCollection(1).Values = "=EntropyLambda!R2C5:R" & numRows - 1 & "C5"
         ActiveChart.Axes(xlValue).Select
         With ActiveChart.Axes(xlValue)
           .MinimumScale = 4
           .MaximumScaleIsAuto = True
```

```
        .MinorUnitIsAuto = True
        .MajorUnitIsAuto = True
        .Crosses = xlAutomatic
        .ReversePlotOrder = False
        .ScaleType = xlLinear
        .DisplayUnit = xlNone
    End With

    ActiveChart.Axes(xlCategory).Select
    With ActiveChart.Axes(xlCategory)
        .MinimumScale = 4
        .MaximumScaleIsAuto = True
        .MinorUnitIsAuto = True
        .MajorUnitIsAuto = True
        .Crosses = xlCustom
        .CrossesAt = 4
        .ReversePlotOrder = False
        .ScaleType = xlLinear
        .DisplayUnit = xlNone
    End With

'get current name of chart and add the trendline                             '(R9.4)
    currentName = ActiveChart.Name
    ActiveWorkbook.Charts("" & currentName).SeriesCollection(1).Trendlines.Add

'trendline details:
    With Charts("" & currentName).SeriesCollection(1).Trendlines(1)
        .Type = xlPower                                                      '(R9.5)
        .DisplayEquation = True
        .DisplayRSquared = False
        'put trendline equation onto entropy lambda sheet
        Worksheets("EntropyLambda").Cells(1, 6).Value = .DataLabel.Text
        .DisplayRSquared = True
    End With

'move trendline label
    ActiveChart.SeriesCollection(1).Trendlines(1).DataLabel.Select
        Selection.Left = 494
        Selection.Top = 198

'format chart for print and change name of chart
    Call formatChartForPrint(currentName)
'(R10.11, R10.12)
    Sheets("" & currentName).Select
    Sheets("" & currentName).Name = "Map Entropy K, K+1"                     '(R9.1)

End Sub 'createMapKK1

'-------------------------------------------------------------------------
'  Subroutine: createLambdaChart
'  Author: Matt Behnke
'  Created: 9/19/01
'  Description: creates a chart based on the lambda calculations.. plots three data
'          series: 1) cumulative entropy, 2) lambda, 3) cum entropy - lambda
'  inputs: none
```

```
'  Outputs: none
'--------------------------------------------------------------------------
Sub createLambdaChart()

  chartName = "EntropyLambdaChart"
  termEnd = CountRows("Entropy Lambda", 3)

  'add chart and set to XY smooth                                            (R7.1)
   Charts.Add
   ActiveChart.ChartType = xlXYScatterSmooth
  'set first data series to use cumulative entropy                           (R7.3)
   ActiveChart.SetSourceData Source:=Sheets("EntropyLambda").Range("$C$2:$C$" & termEnd),
PlotBy:=xlColumns
   ActiveChart.Location Where:=xlLocationAsNewSheet

  'change title, axis labels..
   With ActiveChart
    .HasLegend = True                                                        '(R8.5)
    .HasTitle = True
    .HasAxis(xlValue, xlPrimary) = True
    .HasAxis(xlValue, xlSecondary) = True

    .ChartTitle.Characters.Text = "Entropy (SB) f(Lambda, B)"                 '(R8.1)
    .Axes(xlCategory, xlPrimary).HasTitle = True
    .Axes(xlCategory, xlPrimary).AxisTitle.Characters.Text = "k (Years)"      '(R8.2)
    .Axes(xlValue, xlPrimary).HasTitle = True
    .Axes(xlValue, xlPrimary).AxisTitle.Characters.Text = "Entropy Sk (Bits)" '(R8.3)
   End With

  'change name of dataseries 1                                               '(R7.2)
  'change line style and marker style dataseries 1                           '(R8.8,
R8.9)
   ActiveChart.SeriesCollection(1).Name = "Entropy (Information SH)"
   ActiveChart.SeriesCollection(1).Select
    With Selection.Border
      .ColorIndex = 1
      .Weight = xlMedium
      .LineStyle = xlDot
    End With
    With Selection
      .MarkerBackgroundColorIndex = 1
      .MarkerForegroundColorIndex = 6
      .MarkerStyle = xlTriangle
      .Smooth = False
      .MarkerSize = 9
      .Shadow = True
    End With
    ActiveChart.PlotArea.Select

  'add second data series                                                    '(R7.4,
R7.5)
   ActiveChart.SeriesCollection.NewSeries          'data starts in row 2 column 11
   ActiveChart.SeriesCollection(2).Values = "=EntropyLambda!R2C13:R" & termEnd & "C13"
   ActiveChart.SeriesCollection(2).Name = "Entropy Constant C to relate S_H with S_B
(Lambda)"
```

```vba
        ActiveChart.SeriesCollection(2).Select  'format 2nd data series                    (R8.10,
R8.11)
        With Selection.Border
          .Weight = xlThin
          .LineStyle = xlNone
        End With
        With Selection
          .MarkerBackgroundColorIndex = 50
          .MarkerForegroundColorIndex = 4
          .MarkerStyle = xlSquare
          .Smooth = False
          .MarkerSize = 5
          .Shadow = False
        End With

     'add lambda b series                                                                 '(R7.6,
R7.7)
        ActiveChart.SeriesCollection.NewSeries 'lambda with b    data is in row 2 column 12
        ActiveChart.SeriesCollection(3).Values = "=EntropyLambda!R2C14:R" & termEnd & "C14"
        ActiveChart.SeriesCollection(3).Name = "Lambda with B = 10%"
        ActiveChart.SeriesCollection(3).AxisGroup = 2

        ActiveChart.Axes(xlValue, xlSecondary).HasTitle = True 'set title                 (R8.4)
        ActiveChart.Axes(xlValue, xlSecondary).AxisTitle.Characters.Text = "Lambda = f(ß, u)"

        ActiveChart.SeriesCollection(3).Select 'format 3rd data series                    (R8.12,
R8.13)
        With Selection.Border
          .Weight = xlThin
          .LineStyle = xlNone
        End With
        With Selection
          .MarkerBackgroundColorIndex = xlAutomatic
          .MarkerForegroundColorIndex = xlAutomatic
          .MarkerStyle = xlAutomatic
          .Smooth = False
          .MarkerSize = 5
          .Shadow = True
        End With

     'move legend and adjust chart size                                                   (R8.6,
R8.7)
        ActiveChart.PlotArea.Select
        Selection.Width = 598
        Selection.Height = 395
        ActiveChart.Legend.Select
        Selection.Left = 326
        Selection.Top = 207

     'increase legend size
        ActiveChart.Legend.Select
        Selection.Width = 201

     'remove border and color fill on plot area
```

434

```
      ActiveChart.PlotArea.Select
        With Selection.Border
          .Weight = xlHairline
          .LineStyle = xlNone
        End With
        Selection.Interior.ColorIndex = xlNone

    'remove border on legend
      ActiveChart.Legend.Select
        With Selection.Border
          .Weight = xlHairline
          .LineStyle = xlNone
        End With

      currentName = ActiveChart.Name
      Call formatChartForPrint(currentName)                          '(R8.14,
R8.15)

      Sheets("" & currentName).Select
      Sheets("" & currentName).Name = chartName

    End Sub 'create lambda chart

      '--------------------------------------------------------------------------
      ' Subroutine: formatSheetForPrint
      ' Author: Matt Behnke
      ' Created: 9/19/01
      ' Description: formats the sheet to fit on one page wide (legal size paper)
      '          adds header and footer to each sheet and sets orientation to landscape
      ' inputs: none
      ' Outputs: none
      '--------------------------------------------------------------------------
    Sub formatSheetForPrint()
    'column heading                                                   (R11.3)
      With ActiveSheet.PageSetup
        .PrintTitleRows = "$3:$3"
        .PrintTitleColumns = ""
      End With
      ActiveSheet.PageSetup.PrintArea = "$A$1:$Y$203"
      With ActiveSheet.PageSetup
        .LeftHeader = ""
        .CenterHeader = "&A in &F"                                    '(R11.4)
        .RightHeader = ""
        .LeftFooter = "&D"                                            '(R11.5)
        .CenterFooter = "Page &P of &N"
        .RightFooter = ""
        .LeftMargin = Application.InchesToPoints(0.75)
        .RightMargin = Application.InchesToPoints(0.75)
        .TopMargin = Application.InchesToPoints(1)
        .BottomMargin = Application.InchesToPoints(1)
        .HeaderMargin = Application.InchesToPoints(0.5)
        .FooterMargin = Application.InchesToPoints(0.5)
        .PrintHeadings = False
        .PrintGridlines = True
        .PrintComments = xlPrintNoComments
```

```
            .CenterHorizontally = False
            .CenterVertically = False
            .Orientation = xlLandscape                                '(R11.6)
            .Draft = False
            .PaperSize = xlPaperLegal                                 '(R11.1)
            .FirstPageNumber = xlAutomatic
            .Order = xlDownThenOver
            .BlackAndWhite = False
            .Zoom = False
            .FitToPagesWide = 1                                       '(R11.2)
            .FitToPagesTall = 99
        End With
End Sub 'format sheet for print


'--------------------------------------------------------------------------
'  Subroutine: formatChartForPrint
'  Author: Matt Behnke
'  Created: 9/19/01
'  Description: puts headings and footers on charts. sets to landscape
'  inputs: none
'  Outputs: none
'--------------------------------------------------------------------------
Sub formatChartForPrint(ByVal chartName As String)
    Charts("" & chartName).Select

    With ActiveChart.PageSetup
        .LeftHeader = ""
        .CenterHeader = ""
        .RightHeader = ""
        .LeftFooter = "&D"
        .CenterFooter = ""
        .RightFooter = "&A in &F"
        .LeftMargin = Application.InchesToPoints(0.75)
        .RightMargin = Application.InchesToPoints(0.75)
        .TopMargin = Application.InchesToPoints(1)
        .BottomMargin = Application.InchesToPoints(1)
        .HeaderMargin = Application.InchesToPoints(0.5)
        .FooterMargin = Application.InchesToPoints(0.5)
        .ChartSize = xlFullPage
        .PrintQuality = 600
        .CenterHorizontally = False
        .CenterVertically = False
        .Orientation = xlLandscape
        .Draft = False
        .PaperSize = xlPaperLetter
        .FirstPageNumber = xlAutomatic
        .BlackAndWhite = False
        .Zoom = 100
    End With

End Sub 'format chart for print
'--------------------------------------------------------------------------
'  Function: CountRows
'  Author: ? Revised by: Matt Behnke
'  Created: ?
```

436

```
' Revised: 9/10/01
' Description: Counts the rows in the suppiled worksheet and column number
' inputs:   sheetName - name of the sheet to count the rows in
'        colNum - number of the column to count rows in
' Outputs: number of rows as a double
'-----------------------------------------------------------------------
Function CountRows(ByVal sheetName As String, ByVal colNum As Integer) As Double
On Error Resume Next
Dim currCell As Range, rowNum As Double

  Sheets("" & sheetName).Select

  If IsNumeric(colNum) Then
  Else
    colNum = 1
  End If

  rowNum = 1
  Set currCell = ActiveSheet.Cells(rowNum, colNum)
  Do While currCell.Value <> ""
    rowNum = rowNum + 1
    Set currCell = ActiveSheet.Cells(rowNum, colNum)
  Loop
  CountRows = rowNum - 1
End Function 'CountRows
'-----------------------------------------------------------------------
' Function: CountCols
' Author: ? Revised by: Matt Behnke
' Created: ?
' Revised: 9/10/01
' Description: Counts the rows in the suppiled worksheet and column number
' inputs:   sheetName - name of the sheet to count the columns in
'        rowNum - number of the row to count columns in
' Outputs: number of columns as a double
'-----------------------------------------------------------------------
Function CountCols(ByVal sheetName As String, ByVal rowNum As Integer) As Integer
On Error Resume Next
Dim currCell As Range, colNum As Integer

  Sheets("" & sheetName).Select

  If IsNumeric(rowNum) Then
  Else
    rowNum = 1
  End If
  colNum = 1
  Set currCell = ActiveSheet.Cells(rowNum, colNum)
  Do While currCell.Value <> ""
    colNum = colNum + 1
    Set currCell = ActiveSheet.Cells(rowNum, colNum)
  Loop
  CountCols = colNum - 1
End Function 'CountCols
'-----------------------------------------------------------------------
' Function: cols
```

```
'  Author: Matt Behnke
'  Created: 9/11/01
'  Description: changes column number into a letter.
'  inputs:   columnNumber
'  Outputs: column letter
'-----------------------------------------------------------------------------
Function col(ByVal columnNumber As Integer) As String

  Select Case columnNumber
    Case 1
      col = "A"
    Case 2
      col = "B"
    Case 3
      col = "C"
    Case 4
      col = "D"
    Case 5
      col = "E"
    Case 6
      col = "F"
    Case 7
      col = "G"
    Case 8
      col = "H"
    Case 9
      col = "i"
    Case 10
      col = "J"
    Case 11
      col = "K"
    Case 12
      col = "L"
    Case 13
      col = "M"
    Case 14
      col = "N"
    Case 15
      col = "O"
    Case 16
      col = "P"
    Case 17
      col = "Q"
    Case 18
      col = "R"
    Case 19
      col = "S"
    Case 20
      col = "T"
    Case 21
      col = "U"
    Case 22
      col = "V"
    Case 23
      col = "W"
```

438

```
        Case 24
           col = "X"
        Case 25
           col = "Y"
        Case 26
           col = "Z"
        Case 27
           col = "AA"
        Case 28
           col = "AB"
        Case 29
           col = "AC"
        Case 30
           col = "AD"
        Case 31
           col = "AE"
        Case 32
           col = "AF"
        Case 33
           col = "AG"
        Case 34
           col = "AH"
        Case 35
           col = "AI"
        Case 36
           col = "AJ"
        Case 37
           col = "AK"
        Case 38
           col = "AL"
        Case 39
           col = "AM"
        Case 40
           col = "AN"
        Case others
           col = "Z"
     End Select

  End Function 'col
```

THIS PAGE INTENTIONALLY LEFT BLANK

APPENDIX F INSPEC DATABASE FIELDS

INSPEC records are divided into the following fields, listed in alphabetic order. Highlighted fields are **limit fields**.

AA	Author Affiliation
AB	Abstract
AI	Astronomical Object Indexing
AN	**Accession Number**
AU	Author
AV	Availability
CC	**Classification Codes**
CD	Conference Details
CI	Chemical Indexing
CL	Copyright Clearance Center Code
CO	CODEN
CP	**Country of Publication**
CS	Copyright Statement (*)
DE	Descriptors
DN	Document Number (*)
DOI	Digital Object Identifier (*)
DS	Dissertation Submission Date
DU	Document Collection URL (*)
ED	Editor
IB	ISBN
ID	Identifiers
IS	ISSN
LA	**Language**
MD	Description of Unconventional Medium
MN	Material Identity Number
NI	Numerical Data Indexing
OP	Original Patent Details
PA	Patent Assignee
PD	Patent Details
PF	Patent File Date
PI	Patent Priority Date
PR	Price
PY	**Publication Year**
RF	**Number of References**
RN	Report Numbers
RT	**Record Type**
SC	SICI (*)
SF	**Subfile**
SK	Sort Key
SO	Source
ST	SICI of Translation (*)
SU	Subject Terms (DE and ID)
TI	Title
TL	Translator
TR	**Treatment Codes**
UD	**Update Code**
UR	Universal Resource Locator (*)

(*) This field is for display only; you cannot search in this field.

Figure F-1. **("Fields", 2001) List of INSPEC database fields and descriptions.**

THIS PAGE INTENTIONALLY LEFT BLANK

APPENDIX G LEARNING CURVE

Then analysis of the trends of number of publications (messages) N vs. time step k is developed in an independent approach. The community, macro level, publication data, N_i represents counts of publications (messages) in the partitions (A, B, C, D) coarse-grained bands $N_{A_k}, N_{B_k}, N_{C_k}, N_{D_k}$.

The power form of the learning curve is explored. The learning rates for each band is developed as a performance index as a function of tasks (messages) performed over time steps which is the relationship that is expected in learning. That is that performance improves with the increase in the number of tasks performed. Then, in a stepwise fashion, entropy is introduced into the learning curve equation, showing how the complexity of the messages being processed in a technology transition task affects the performance index. This is then related to the two-dimensional map of a dynamical system.

1.   Capacity

We will compute the organizational capacity in a band as the number of messages processed on average over the time steps to date. We look at an organization production of messages. The organization messages produced are allocated to the number of authors in order to get the average number of messages per author per time step. This is done by organizational bands. We observe the apparent capacity of the organizations in the "A" band (the best performers by cumulative messages produced) and allocate it to the number of authors. Now we have what could be considered, the property of the best capacity available in the channel.

In the entropy learning curve model, we use this as the best performance we might expect. It is well accepted that an individuals performance, in terms of tasks per

unit time, improves through learning as a function of the number of times the task is performed. (Mazur 1978) (Newell 1981). So the more times, *N,* that a task is performed, the tasks per time step performance index improves. We observed this in these models as well. An important part of this research is to develop the relationship between tasks performed in a time step by an author (on average) and the complexity of the message.

To bridge the gap between communication theory and capacity of human performance an analogy is made between, a human accepting input and generating output and a communication system. This is seen as the overlap in a Venn diagram. The input variance is represented by the circle to the left, and the output variance is the circle to the right, and at the intersection is the amount of transmitted information. Miller (Miller 1956) suggests that an individual is a communication channel. He states for a human, "when we increase the amount of input information, the transmitted information will increase at first and will eventually level off at some asymptotic level." He indicated that this is the channel capacity of the observer, the human. We also see that there is a capacity and that the performance levels off. A further discussion on this is found in section Appendix G Learning Curve, (p443).

2.   Pressure

Let's establish a conceptual framework for pressure. Imagine a physical system with a channel made up of a number of garden hoses, with each hose having a finite cross sectional area. We can denote pressure in terms of pounds per cross sectional area, or pounds[1] per square inch say. If the hoses in a band were treated on average as the same size, we could indicate pressure in pounds per hose. This could be stated in pounds per channel. We might say the pressure is some force measure per node if the channel was made up of a collection of nodes strung together in a kind of a graph. All the terms in a given state and node ensemble is state space represent the volume.

---

[1] In this illustration, we are using the engineering sense of pounds mass. Recall that force is proportional to the second derivative of a length, a step *l*, with respect to time. $F \propto d^2l/dt^2$. The important piece to notice is in the math here, not whether we have the right units on the force or not. $\propto$, the proportionality constant is mass in Newton's equations. For our purposes, let's not think in terms of force and mass which is related to gravitation and our physical world, but look at the mathematical meaning and see the proportionality constant. For convenience, we will call it a mass, actually a probability mass.

We almost have enough information theory to understand the models. Consider the entropy as a representation of the terms in a vocabulary, which are available to the researchers in a time step. A researcher reaches into the pool of messages, which are constituted by terms. We can compute entropy contribution of a term in a given time step as a function of *p(x)* for the term. Summing all of the terms' entropy contribution, we have the entropy at time step *k*.

A_Band Productivity In Pubs (Cum over k)

A_Band Productivity Index (Cum over k)

Learning Curve -- A_Band (Mean and Capacity)
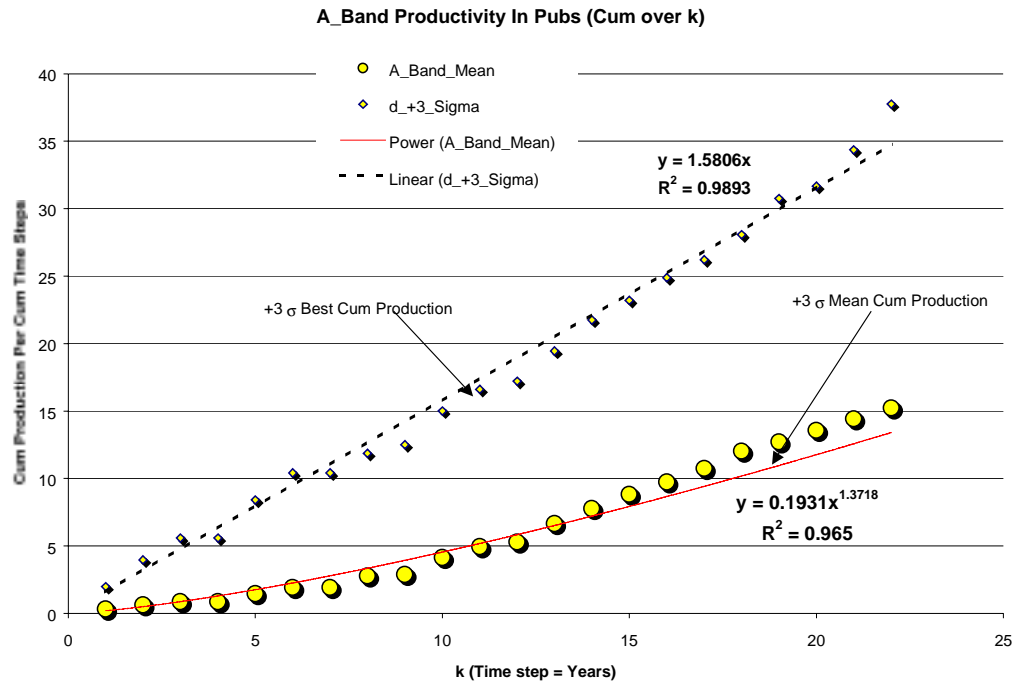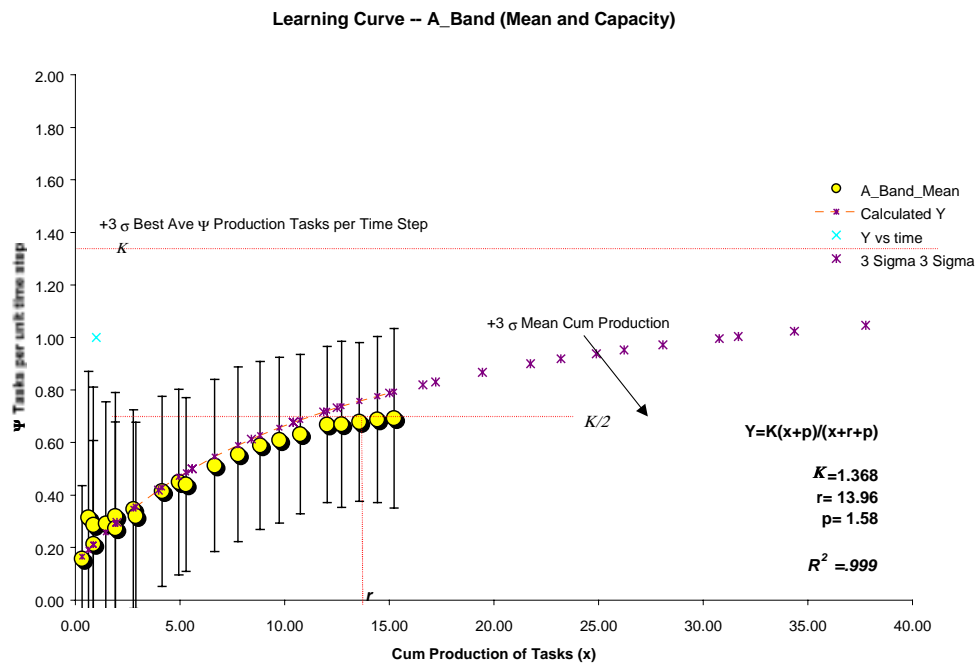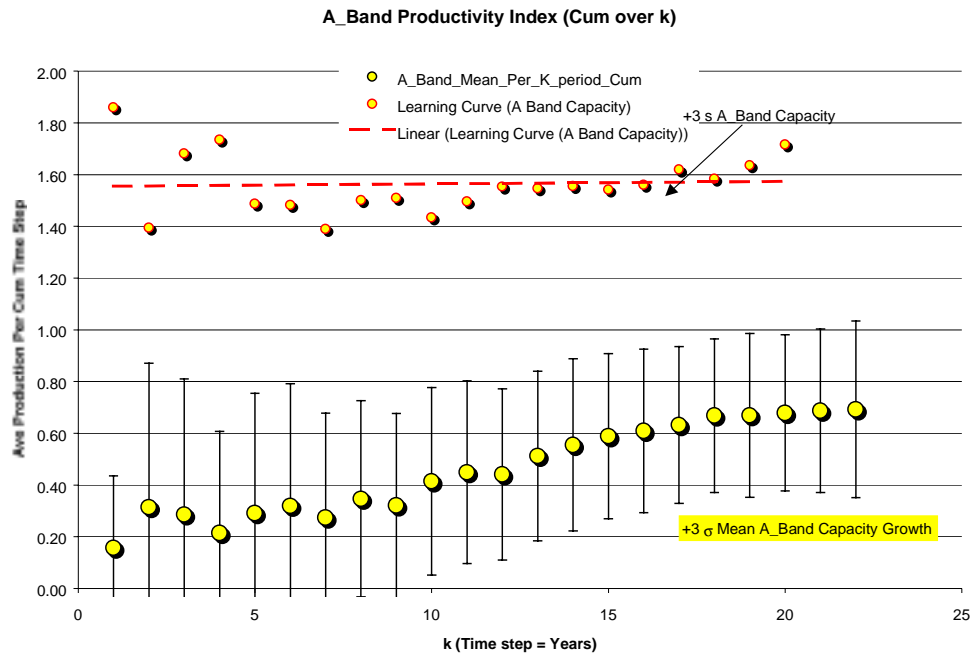
Since we have the affiliated publisher information, we can find a performance indicator per capita of the set for $\overline{X}_i$ the in the organization band as

described in the section on technology transfer system elements. At each time step, we can determine the maximum (on average) capacity per capita in a band. This will yield a set of capacity productivity curves representing the community learning in bands. This approach yields a learning curve, which is an average for the set of performers measured in the data set. This then has the individual based description of learning within a population of learners. An individual based approach views the organization as a population of learners, with organizational learning is a sum of the individual behaviors. This establishes criteria for the performance indicator for capability and experience in the N dimension. The next time step the process is repeated to provide N_k. This is repeated for n time steps, where n is the upper bound over the range of data being examined. This builds a moving distribution with a time varying performance indicator criteria.

While it is tempting to relate the Rogers 1983 adopter profile, we can not do this directly with the data as presented. If the performers are ordered in the time step when they first appear, then the true innovators, early adopters, early majority, and late majority can be identified. We also do not expect to find the laggards publishing.

The data must therefore include the term count, entropy by term (a calculated value) and publication rate for author and affiliation allocated to an accession number (AN). The accession numbers are allocated to bins. These can be a year, a month (year AN ranges divided by 12) or weekly (year AN numbers divided by 50, since there are 50 updates to the IEEE database per year). While the time step k, is set by the bin size the interval of meaning is k-c.   Where c is the number of time step that improves convergence of a feedback model. For example, if the bins are weekly, we take a year offset to publish, request clarification and another year (from a publishing cycle) to have the request for clarification be received in a published message.

Nembhard and Uzumeri (Nembard 2000) studied twelve learning curves. They found exponential and hyperbolic learning curves are the best suited for mixed perceptual and motor learning.   The curves analyzed are discussed here for reference.   These represent the major contributions of the underlying learning curve research.

They compared models for aggregation, and individual learning. Aggregation implies that you can sum up individual learning and have a representation of organizational learning. Although it is possible to derive lower level information from aggregated data, it is generally difficult to disaggregate organizational level learning into smaller organizational units where the workplace interventions and changes are actually implemented. It is also difficult to separate the learning effects from the effects of other internal and environmental effects (Nembhard 2000). They note that organizational (aggregate) learning curves are best used for measuring organizational improvements over time. They also looked at models that would be appropriate for both individual and aggregation, referred to as the combined model. These are summarized below with the number in parenthesis indicating the goodness of the model as found by Nembhard.

- Aggregation models which permit taking learning measures at the individual level and aggregations of those measures represent organizational level reality)

    - 3) DeJong's learning formula (DeJong 1957)

    - 4) Stanford *B* model (Asher 1956)

    - 5) Log linear (Wright 1936)

    - 6) S-Curve (Carr 1947)

    - 10) Levy's function (Levy 1965)

- Individual models permitting measures at the individual level, but not necessarily being able to aggregate to a meaningful organizational aggregate.

    - (2) exponential functions (two and three parameter) (Mazur 1978)

    - (1) hyperbolic functions (two and three parameter) (Mazur 1978)

- Combined models permitting accounting for empirical data observations in learning data.

    - (8) Pegels' function (Pegels 1969)

    - (11) Knecht's model (Knecht 1974)

    - (2) exponential functions (two and three parameter) (Mazur 1978)

It is useful to present some of the basics of the power law.

$$T = BN^{-\alpha} \tag{G.1}$$

or in log-log form

$$\log(T) = \log(B) - \alpha \log(N) \tag{G.2}$$

Where N is the number of trials and T is the time it takes to perform a task, $-\alpha$ is the slope and B is the offset reflecting prior experience or trials. Looking at this in terms of the rate of local learning, dT/dN, we see

$$\frac{dT}{dN} = -\alpha BN^{-\alpha-1} \tag{G.3}$$

We know that one form of learning is exponential. It can arise from any mechanism that is completely local. Therefore, if there is something that learns on each local part of performance, independent of any other part, then the change in T (the sum of the changes to each part of T) is proportional to T:

$$\frac{dT}{dN} = -\alpha T \tag{G.4}$$

$$T = Be^{-\alpha N} \tag{G.5}$$

Comparing this differential form with the power law, shows that the power-law is like exponential learning, in which the instantaneous rate α' decreases with N, that is,

$$\frac{dT}{dN} = -\alpha' T \tag{G.6}$$

where $\alpha' = \alpha/N$

The three parameter hyperbolic is given here in more detail since the variables in can seen from this form. This is also the best model for describing learning across

populations of individuals. The plots of the hyperbolic and exponential ignore prior learning $p=0$ for

$$\psi = \kappa \left(\frac{x+p}{x+p+r} \right)$$

such that $\psi, \kappa, p, x \geq 0,$ and (G.7)

$p+r \rangle 0$

ψ is the measure of work performance, and x is the amount of cumulative work in units of time or number of trials (messages in the case of this research). The parameter k provides an estimate of the asymptotic limit or maximum performance level that can be expected when all learning has been completed. The upper bound on κ (kappa) comes from a distribution of workforce performance. In this research, we assume the originator of the technology (the advocate) could do. For example, assume the SEI is the most prolific on a technology in a given time step. So if the SEI publishes u_{SEI} messages in a given time step, then $\kappa =1/ u_{SEI}$. Parameter r is the cumulative production required in order to attain an output level of $\kappa/2$ and represents the rate at which productivity converges toward κ. Small values of r indicate that learning occurs rapidly *relative* to κ. The value of r may also be small if the publishing unit reaches steady state limit. This can happen quickly with prior experience. p represents the individual performing activity's accumulated prior experience on a time or a cumulative messages basis. The prior experience may be acquired from the work on similar tasks (messages) and interpreted as the point on the learning curve where the unit is resuming the learning process.

Note that the denominator $(x+p+r)$ must be non zero. Since cum tasks or time is positive, this implies $p>-r$. The model's first and second derivatives are:

$$\frac{\partial \psi}{\partial x} = \frac{kx}{(x+p+r)^2} \qquad\qquad (G.8)$$

$$\frac{\partial^2 \psi}{\partial x^2} = \frac{-2kx}{(x+p+r)^3} \qquad\qquad (G.9)$$

In order to illustrate the general shape of the learning curves the hyperbolic and exponential forms are plotted in Figure G-1. Figure G-1 is a plot of a three parameter hyperbolic learning curve with one parameter *p* for prior learning set to 0. Figure G-2 is a plot of a three parameter exponential learning curve, also with curve with one parameter *p* for prior learning set to 0. The parameter *p>0* shifts the curves to the left by the amount of *p,* prior tasks performed
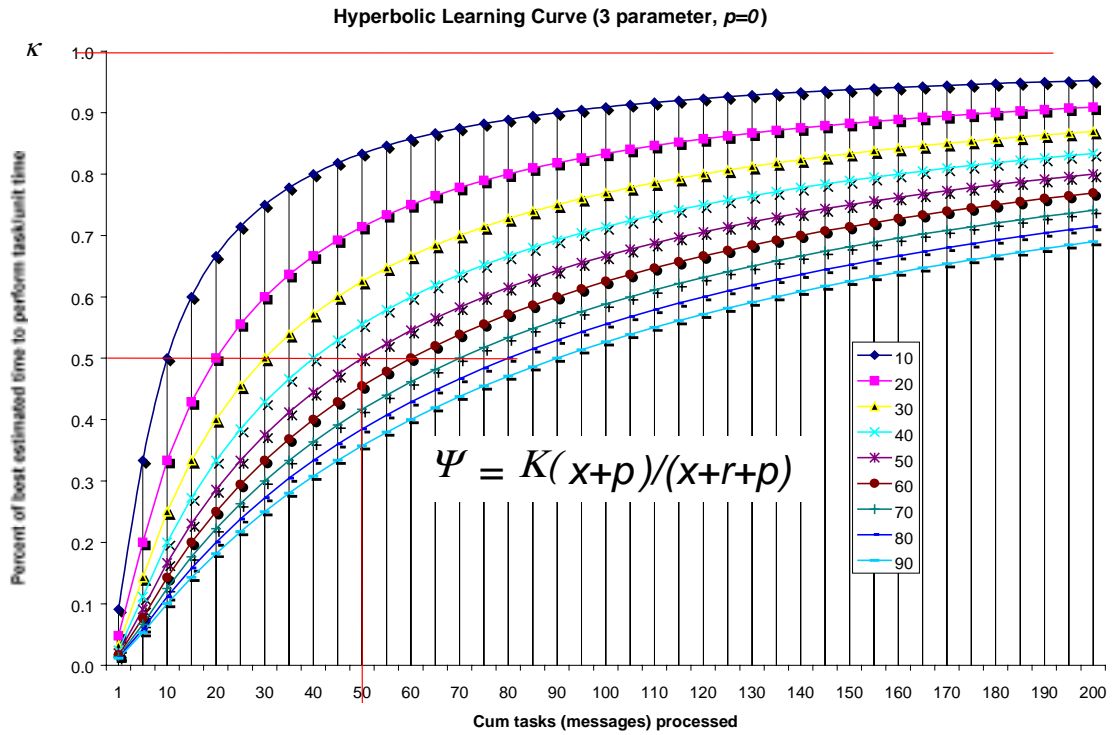
Figure G-1 Hyperbolic Learning Curve (3 Parameter).

(Source: after Nembhard 2000)
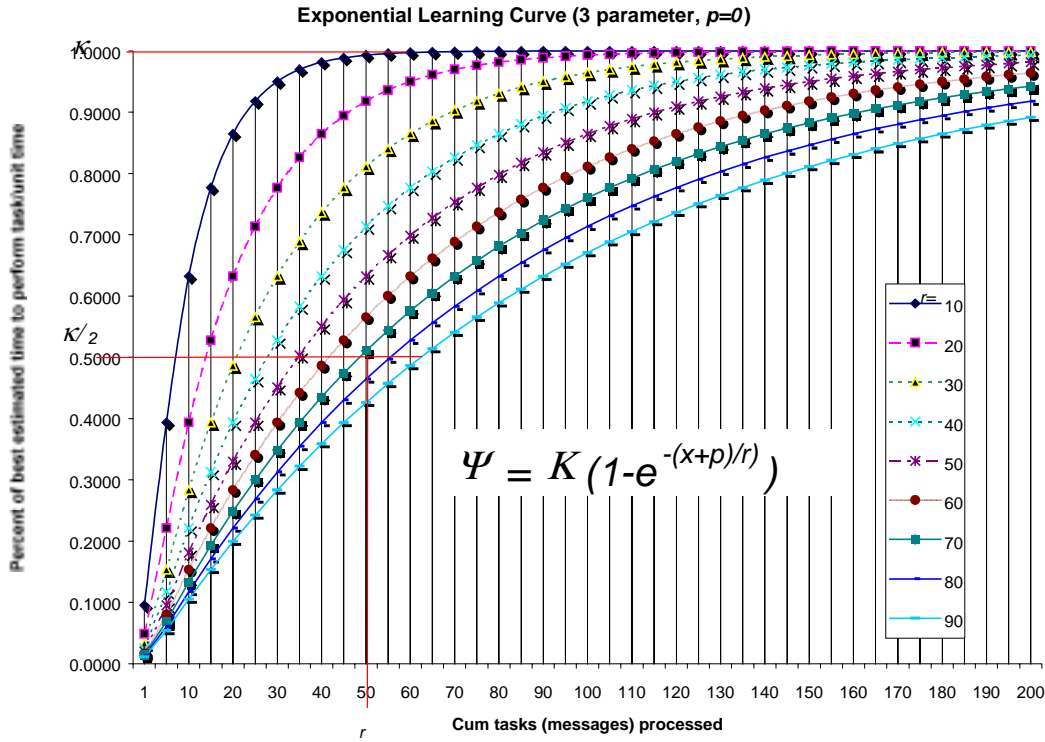
Figure G-2   Exponential Learning Curve (3 Parameter).

(Source: after Nembhard 2000)

3.   Trials and Time Relationship

The basic law of practice is of the form of a power law (G.1), and has also shown below in log-log form (G.2)

The form of the law of practice is performance time (*T*) as a function of trials (*N*). However, trials are simply a way of marking the temporal continuum (*t*) into intervals, each one performance-time long. Since the performance time is itself, a monotone decreasing function of trial number, trials (*N*) becomes a nonlinear compression of time (*t*). It is important to understand the effect on the law of practice by viewing it in terms of time or in terms of numbers of trials.

The control algorithm has the number of messages processed without requiring a request for feedback as $f(x_k)$. This is the number of messages (trials) input at time step *k*. The fundamental relationship between time and trials is obtained as follows:

$$t(N) = T_0 + \sum_{i=1}^{N} T_i = T_0 + \sum_{i=1}^{N} Bi^{-\alpha} = T_0 + B - \sum_{i=1}^{N} i^{-\alpha}$$

(G.10)

T_0 is from the arbitrary time origin to start the first trial. This equation cannot be inverted explicitly to obtain the expression for $N(t)$ that would permit the basic law (Equation 0-1) to be transformed to yield $T(t)$. Instead, we proceed indirectly by means of differential form. From we obtain

$$\frac{dT}{dN} = T$$

(G11)

Using the following integral formulation

$$\frac{d}{dz} \int_a^z f(x)dx = f(z)$$

Now starting with the power law in terms of trials we find

$$\frac{dT}{dN} = \frac{dT/dN}{dt/dN} = \frac{-\alpha T/N}{T} = \frac{-\alpha}{N}$$

(G.12)

But from (G.1) we get

$$N = \left(\frac{T}{B}\right)^{-1/\alpha}$$

(G.13)

$$N = \left(\frac{Ce^{\frac{-1}{B}t}}{B}\right)^{-1/\alpha} = \left(\frac{C}{B}\right)^{-\frac{1}{\alpha}} e^{\frac{1}{B\alpha}t}$$

(G.14)

where C^i is $\left(\dfrac{C}{B}\right)^{-\frac{1}{\alpha}}$

(G.15)

When $\alpha = 1$

$$\frac{dT}{dt} = -\frac{1}{B}T$$

(G.16)

By solving the differential equation, we get

$$T = Ce^{-\frac{1}{B}t} \qquad\qquad\qquad\qquad \text{(G.17)}$$

When $\alpha \neq 1$ we have a polynomial that we can integrate, where C is an arbitrary constant of integration and if the origin and scale of t is adjusted properly and get

$$T^{-(1-\alpha)/\alpha} = (1-\alpha)B^{-1/\alpha}t + C \qquad\qquad \text{(G.18)}$$

So, we can obtain the trials power law re-expressed in terms of time:

$$\frac{dT}{dt} = -\alpha B^{-1/\alpha}T^{1/\alpha} \qquad\qquad\qquad \text{(G.19)}$$

Rearranging $\dfrac{dT}{T^{1/\alpha}} = -\alpha B^{-1/\alpha}dt$ and integrating both sides, we get

$$\int_0^t \frac{1}{1-\frac{1}{\alpha}}dT^{1-1/\alpha} = \int_0^t -\alpha B^{-1/\alpha}dt \qquad\qquad \text{(G.20)}$$

$$\frac{\alpha}{\alpha-1}T^{\frac{\alpha-1}{\alpha}} + C_1 = -\alpha B^{-1/\alpha}t + C_2 \qquad\qquad \text{(G.21)}$$

adjust the constants of integration equal to 0

$$\frac{\alpha}{\alpha-1}T^{\frac{\alpha-1}{\alpha}} = -\alpha B^{-1/\alpha}t \qquad\qquad\qquad \text{(G.22)}$$

rearranging we get

$$T^{\frac{\alpha-1}{\alpha}} = (1-\alpha)B^{-1/\alpha}t \qquad\qquad\qquad \text{(G.23)}$$

$$T = \left[(1-\alpha)B^{-1/\alpha}t\right]^{\frac{\alpha-1}{\alpha}} t^{\frac{\alpha-1}{\alpha}} \qquad\qquad \text{(G.24)}$$

which has a constant as the coefficient and we can write it as B'

$$T = B't^{\frac{\alpha-1}{\alpha}} \qquad\qquad \text{for } \alpha \neq 1 \qquad\qquad \text{(G.25)}$$

This is similar to (1.26) with N given as a function of time. Rewriting we see

$$T = B'N^{-\alpha} \qquad\qquad\qquad\qquad \text{(G.26)}$$

Therefore, we now have two possibilities for T

$$T = \begin{cases} B't^{-\alpha\frac{1}{1-\alpha}} & \alpha \neq 1 \\ Ce^{-\frac{1}{B}t} & \alpha = 1 \end{cases} \qquad \text{(G.27)}$$

THIS PAGE INTENTIONALLY LEFT BLANK

APPENDIX H ANNOTATED BIBLIOGRAPHY TECHNOLOGY TRANSFER

TECHNOLOGY TRANSITION ANNOTATED BIBLIOGRAPHY

This annotated bibliography contains complete bibliographic citations of most of the relevant technology transfer literature for software engineering. It does not include experience reports, or case studies in general. Some from IBM, HP and a few other notable (Cleanroom) studies are included due to the extensive study on transitioning those technologies.

Most of the citations include a category and keywords. At the end of this annotated bibliography is the 1988 paper by Przybylinski. This paper provided a data set to explain entropy in Chapter III. The current bibliography will be updated and posted on the SEI web site in 2002.

(Abetti 1995) Abetti, Pier A. and Robert W. Stuart. "Entrepreneurship and Technology Transfer: Key Factors in the Innovation Process," in Donald L. Sexton and Raymond W. Smilor (editors), *The Art and Science of Entrepreneurship,* chapter 7, pages 181-210. Ballinger Publishing Company, Cambridge MA, 1985.

Category: technology transfer

Key Words: innovation, entrepreneurship

Abstract/Summary: *This paper provides a linear model of the technological innovation process, with two case studies (non-impact magnetic printer and extra-high voltage transformers) defined in terms of that model. The cases focus on the importance of the different roles in the innovation process, e.g., gatekeepers, champions, etc.*

Referenced by (Przbylinski 1988)

(Adrion 1994) Adrion-WR; McOwen-P, "A Three-Pronged Strategy For Technology Creation, Transfer And Absorption," in Levine, Linda, ed., proceedings of the IFIP TC8 Working Conference on Diffusion, Transfer and Implementation of Information Technology, Software Engineering Institute, Carnegie Mellon Institute, Pittsburgh, PA, North Holland, Amsterdam,, London, New York, Tokyo, 1994.

SOURCE: IFIP-Transactions-A-(Computer-Science-and-Technology). vol.A-45; 1994; p.309-20

ABSTRACT: *The Computer Science Department of the University of Massachusetts, Amherst has developed a strategy for research, development, industrial interactions and technology transfer called the "Three Pronged Strategy (TPS)". The principal components within the Three-Pronged Strategy are: continuing programs of education and fundamental research in computer science within the Computer Science Department of the University of Massachusetts, Amherst; a program of focused, or "problem-driven" basic research within the Center for Real-Time and Intelligent Complex Computing Systems (CRICCS); and a program of applied research and development and technology transfer within the Applied Computing Systems Institute of Massachusetts (ACSIOM). In this report, we discuss the motivation and development of the TPS and our experiences to date. We describe each of the components of our strategy and suggest how these might be adapted to other environments.*

REF: 0

(Allen 1977) Allen, Thomas John. *Managing The Flow Of Technology: Technology Transfer And The Dissemination Of*

Technological Information Within The R&D organization., The MIT Press, Cambridge, MA, 1977.

Category: communication

Key Words: communication, dissemination

Abstract/Summary: *Allen's book summarizes his detailed study of communication processes and their impact on the technology development process in a R&D environment. His work has implications on topics such as technical publishing, human resource development and office design.*

Referenced by (Przbylinski 1988)

(Ardis 1994) Ardis, M.A.; Furchtgott, D.G., "Research and development: differences are barriers to transfer," in Levine, Linda, ed., proceedings of the IFIP TC8 Working Conference on Diffusion, Transfer and Implementation of Information Technology, Software Engineering Institute, Carnegie Mellon Institute, Pittsburgh, PA, North Holland, Amsterdam,, London, New York, Tokyo, 1994.

SOURCE: IFIP-Transactions-A-(Computer-Science-and-Technology). vol.A-45; 1994; p.245-7

ABSTRACT: *We have discovered several differences between research and development that frustrate attempts to introduce new software technology into development. For each of these differences we have found strategies that either reduce the difference or mitigate its effects.*

REF: 1

(Bailey 1982) Bailey, Claudia Lynn. "Technology Transfer: A Compilation of Varied Approaches to the Management of Innovation," Master's thesis, Naval Postgraduate School, December, 1982.

Category: technology transfer

Key Words: innovation, technology management

Abstract/Summary: *This masters thesis from the Naval Postgraduate School provides abstracts of many technology transfer references available during that period.*

Referenced by (Przbylinski 1988)

(Barrett 1984) Barrett, Edgar and Donna Bergstedt. "The System Texas Instruments Developed To Manage Innovation," *International Management* 0:81-87, May, 1984.

Category: innovation

Key Words: technology management, technology planning, strategic planning

Abstract/Summary: *This article was condensed from a case study prepared by Professor Barrett from Southern Methodist University. It details the Objectives, Strategy and Tactics (OST) system, a layered planning system in place at Texas Instruments. OST includes (1) a hierarchical goal system; (2) dual responsibility (strategy development and operations) of line management; and (3) analysis of the impacts of a matrix organization on these strategic and operating modes. Goals flow from high level business objectives to strategies to Tactical Action Programs (TAPs), where they are implemented on the business unit level. About 75% of TI's managers wear both strategic and operating which TI believes forces them to do long-range thinking. (The full case is available from Case Publishing,* 46 *Glen Street, Dover, Massachusetts,* 02030.)

Referenced by (Przbylinski 1988)

(Bass 1994) Bass L; Soule A, "Technology Transition Of User Interface Management Systems," in Levine, Linda, ed., *Proceedings of the IFIP TC8 Working Conference on Diffusion, Transfer and Implementation of Information Technology,* Software Engineering Institute, Carnegie Mellon Institute, Pittsburgh, PA, North Holland, Amsterdam,, London, New York, Tokyo, 1994.

SOURCE: IFIP-Transactions-A-(Computer-Science-and-Technology). vol.A-45; 1994; p.357-68

ABSTRACT: *This paper presents a case study of the transition efforts associated with an advanced user interface tool. The tool (Serpent) was well received scientifically, leading to efforts to influence the standards community, to commercialize a Serpent product, and to formulate a special purpose consortium. The results of these efforts are reported.*

REF: 13

(Bayer 1989) Bayer, Judy and Melone, Nancy, "A Critique of Diffusion Theory as a Managerial Framework for Understanding Adoption of Software Engineering Innovation," 0164-1212/89 IEEE, pp. 161-166, 1989.

REF: 13

(Besselman 1994) Besselman-J., "Position Statement On Software Process Innovations And Informal Organizational Networks," in Levine, Linda, ed., proceedings of the IFIP TC8 Working Conference on Diffusion, Transfer and Implementation of Information Technology, Software Engineering Institute, Carnegie Mellon Institute, Pittsburgh, PA, North Holland, Amsterdam,, London, New York, Tokyo, 1994.
SOURCE: IFIP-Transactions-A-(Computer-Science-and-Technology). vol.A-45; 1994; p.321-5

ABSTRACT: *The Software Engineering Institute (SEI) at Carnegie Mellon University spawned the software process improvement industry about six years ago (1988), with their initial version of the capability maturity model (CMM) for software. This position statement outlines the author's research agenda after reviewing many software development organizations over the last few years. Most software development organizations are engaged in some type of program of software process improvement. The inattention paid to the informal organization is identified as a weakness in many of these software process improvement programs. Additionally, a decomposition of what constitutes a software process innovation is presented as a precursor for developing a research model of process innovations covering all software development activities.*
REF: 16

(Bihari 1994) Bihari, T.E.; Varner, M.O., "Practical Issues In Information Technology Transfer," in *Diffusion, Transfer and Implementation of Information Technology*, in Levine, Linda, ed., proceedings of the IFIP TC8 Working Conference on Diffusion, Transfer and Implementation of Information Technology, Software Engineering Institute, Carnegie Mellon Institute, Pittsburgh, PA, North Holland, Amsterdam,, London, New York, Tokyo, 1994.
SOURCE: IFIP-Transactions-A-(Computer-Science-and-Technology). vol.A-45; 1994; p.369-72

ABSTRACT: *Adaptive Machine Technologies, Inc. (AMT) is an engineering research and product development company located near the Ohio State University (OSU). AMT's strengths are primarily in the areas of software and electrical engineering. Since 1984, they have been working with OSU personnel on various projects. Over the last five years (1989-94), AMT has broadened its line of business to include commercial product development, in partnership with other companies and as contractors. AMT frequently works at the boundary between university research and commercial product development. In that position, they have witnessed and been involved in a number of projects that fall under the umbrella of "university-industry technology transfer". Some were official programs but many others consisted of general cross-fertilization between academics and practitioners working in the same application domains. For several years, AMT has been working with the OSU Center for Mapping (CFM) on projects in the GPS/GIS area. CFM collaborates with private sector companies like AMT in an attempt to marry the intellectual capital of the university with the market discipline of the private sector. The CFM is an interesting place to study technology transfer because, unlike the university, their sole mission is to transfer technology. The authors present some general observations and suggestions, based on experiences with university -industry technology transfer at AMT and the CFM.*
REF: 1

(Bikson 1985) Bikson, Tora K., Catherine Stasz and Donald A. Mankin, ."Computer-Mediated Work: Individual and Organizational Impact in One Corporate Headquarters*,"* Final Report R-3308-OT A, Rand, November, 1985.
Abstract/Summary: *This Rand study focused on technology characteristics that enhanced the adoption of office automation technologies.*
Referenced by (Przybylinski 1988)

(Borton 1994) Borton, J..M.; Brancheau,J.C., "Does An Effective Information Technology Implementation Process Guarantee Success?" in *Diffusion, Transfer and Implementation of Information Technology*, in Levine, Linda, ed., *Proceedings of the IFIP TC8 Working Conference on Diffusion, Transfer and Implementation of Information Technology*, Software Engineering Institute, Carnegie Mellon Institute, Pittsburgh, PA, North Holland, Amsterdam,, London, New York, Tokyo, 1994.
SOURCE: IFIP-Transactions-A-(Computer-Science-and-Technology). vol.A-45; 1994; p.159-78

ABSTRACT: *A model of the IT adoption and implementation process is described. The model integrates empirical information system (IS) research with concepts from three theories originally developed in referent disciplines. The model is used to guide a longitudinal case study through sixteen months of qualitative and quantitative data collection. A qualitative analysis of the data is presented describing the implementation process using a temporal (chronological) format. The analysis shows that a strong implementation process within a supportive environment can overcome weaknesses indicated by some of the implementation factors. In addition, the effect of the interaction of factors within and among the stages of the process is clarified, and the cyclical nature of the implementation process is validated in this research context. This study makes two primary contributions to IS research and practice. First, the study demonstrates that a longitudinal research design combined with a mixed quantitative/qualitative data collection approach can provide a rich base of data to use in examining the IT adoption and implementation process. Second, the research provides support for the development of a theory-based model to guide managers in the planning and control of new IT installations.*

REF: 24

(Brownswood 1994) Brownswood L., "Applying Technology Transition In Large Software Organizations," in Levine, Linda, ed., *Proceedings of the IFIP TC8 Working Conference on Diffusion, Transfer and Implementation of Information Technology*, Software Engineering Institute, Carnegie Mellon Institute, Pittsburgh, PA, North Holland, Amsterdam,, London, New York, Tokyo, 1994.
SOURCE: IFIP-Transactions-A-(Computer-Science-and-Technology). vol.A-45; 1994; p.373-6

ABSTRACT: *The author profiles consumer organizations that successfully transition significant software engineering technologies. The success elements are derived from the author's experience supporting seven software organizations which develop and maintain large software systems in the general command and control application domain. These organizations included three United States government contractors, three government contractors in Europe and Australia, and a United States government agency. The technologies these organizations have attempted to transition include software engineering, reuse, object-oriented technology, Ada, computer aided software engineering tools, software measurement programs, and continual process improvement. The process maturity was typical for software organizations of the late 1980's and early 1990's. Most organizations had some level of defined process, although the formality of definition and usage varied.*
REF: 0

(Buxton 1991) Buxton, J.N. and Malcolm, R., *Software Technology Transfer*, 17-23.
Category: Transferring of Technology Between Businesses
Key Words: Technology, Participation, Complex
Abstract/Summary: *Software technology transfer is long and complex between businesses. There are two aspects for any technology to be transferred. First it must be possible to estimate its value in the client organization and second the client organization must be mature and understand the use of the technology. The process of transferring requires the participation of many people (i.e. suppliers, management, gatekeeper, workers, etc.), throughout many unbroken phases (awareness of needed technology, decision making, and adaptation for use), otherwise the outcome will not satisfy the client organization.*

REF: 8

(Childers 1986) Childers, Terry L., "Assessment of the Psychometric Properties of an Opinion Leadership Scale," *Journal of Marketing Research ,* XXIII pp.184-188, May, 1986.
Category: innovation
Key Words: opinion leader, diffusion of innovations
Abstract/Summary: *The concept of opinion leadership is central to the study of the diffusion of innovations. In this article, the author discusses existing efforts to develop a tools for measuring opinion leadership. The paper goes on to describe a study in which a modified opinion leadership scale (i.e., based on the King and Sommers self-designating scale) is shown to have higher internal consistency reliability.*
Referenced by (Przbylinski 1988)

(Christian 1994) Christian-JT; Eward, M.M., "Transferring Software Engineering Technology: The Software Productivity Consortium Experience," in Levine, Linda, ed., *Proceedings of the IFIP TC8 Working Conference on Diffusion, Transfer and Implementation of Information Technology,* Software Engineering Institute, Carnegie Mellon Institute, Pittsburgh, PA, North Holland, Amsterdam,, London, New York, Tokyo, 1994.

SOURCE: IFIP-Transactions-A-(Computer-Science-and-Technology). vol.A-45; 1994; p.377-80

ABSTRACT: *In 1991, the Software Productivity Consortium (the Consortium) rapidly expanded usage of Consortium products by its member companies from fewer than 10 to nearly 100 uses. The Consortium accomplished this by adopting a view of technology transfer as a people-to-people activity, a contact sport. Engaging in this contact sport requires applying a matrix approach to transferring technology that is geared to meeting both common problems of the member companies and unique, individual information and support needs of member company staff. The matrix approach transfers each technology through a diverse set of products and services, cooperative interactions with all member company staff levels, and internally-set expectations for transfer performance and product quality.*

REF: 0

(Clapp 1988) Clapp, Judith, "Government/industry interaction in Ada software engineering tool technology transfer", TH0218-8/88 IEEE p 67-69

Category: Ada Software Technology Transfer

Key Words: Ada, program managers, government, standard interface, compiler

Abstract/Summary: *The government funded the design of Ada ten years ago when no other language was found suitable. The government made Ada required for certain systems and made it difficult for program managers to obtain waivers. To counteract the risk of tools not operating correctly the government has a validation process for the compiler. In conclusion, the transfer has been difficult partly because Ada was forced in through mandates. Risk reduction and feedback is necessary and the link to the technology transfer.*

REF: 0

(Cohen 1994) Cohen, Wesley M. and Levinthal, Daniel A., "Fortune Favors the Prepared Firm:, *Management Science*, Vol. 40., NO. 2, February 1994.

REF: 52

(Cohn 1980) Cohn, Steven F. and Romaine M. Turyn. "The Structure of the Firm and the Adoption of Innovations," *IEEE Transactions on Engineering Management* EM-27(4):98-102, November, 1980.

Abstract/Summary: *This paper describes the impact of organizational structure on innovativeness. Their hypotheses were that adoption varies directly with firm complexity and inversely with centralization and formalization. It contains a number of references to other work in this area.*

Referenced by (Przbylinski 1988)

(Creighton 1972) Creighton, J. W., J. A. Jolly, and S. A. Denning, "Enhancement of Research and Development Output Utilization Efficiencies: Linker Concept Methodology, in the Technology Transfer Process," *Scientific, Interim* AD-756 694, Naval Postgraduate School, June, 1972.

Category: technology transfer

Key Words: adoption, innovation, linker

Abstract/Summary: *Creighton et al studied the characteristics of potential technology adopters and their organization to build a regression model of technology transfer process. Its variables consider innovation, motivational and communication aspects.*

Referenced by (Przbylinski 1988)

(Culver 1994) Culver, Lozo K, "Process engineering support for technology transfer: strategy and experiences," in Levine, Linda, ed., proceedings of the IFIP TC8 Working Conference on Diffusion, Transfer and Implementation of Information Technology, Software Engineering Institute, Carnegie Mellon Institute, Pittsburgh, PA, North Holland, Amsterdam,, London, New York, Tokyo, 1994.

SOURCE: IFIP-Transactions-A-(Computer-Science-and-Technology). vol.A-45; 1994; p.327-31

ABSTRACT: *A software engineering process group (SEPG) can speed the transfer of technology to software development*

organizations. By identifying stable technology that addresses the critical software development needs of an organization, the SEPG can reduce the costs and risks of adopting innovations. The SEPG can also represent the software development needs of the development organization to technology providers in order to promote work focused on solving key development challenges.

REF: 1

(Damsgaard 1994)   Damsgaard, J.; Rogaczewski. A.; Lyytinen X., "How Information Technologies Penetrate Organizations: An Analysis Of Four Alternative Models" in *Diffusion, Transfer and Implementation of Information Technology*, in Levine, Linda, ed., proceedings of the IFIP TC8 Working Conference on Diffusion, Transfer and Implementation of Information Technology, Software Engineering Institute, Carnegie Mellon Institute, Pittsburgh, PA, North Holland, Amsterdam,, London, New York, Tokyo, 1994.

SOURCE: IFIP-Transactions-A-(Computer-Science-and-Technology). vol.A-45; 1994; p.1 –21

ABSTRACT: *We analyze investigations to explain information technology penetration processes. A framework is presented which serves as a common background for exploring four IT penetration models discussed in the literature. The framework strives to unify theoretical accounts to explain IT penetration processes by recognizing six major issues which need to be addressed in any model seeking to explain IT diffusion. These are: penetration level identification criteria, qualitative differences between levels, explanative content of the model, items of penetration, assumed causal structure and underlying theory. The framework is applied to analyze the following four IT penetration models: Nolan's stage theory (1973), Attewell's IT diffusion model (1992), Gurbaxani et al.'s institutional model (1990), and Lyytinen's transaction cost based model (1991). The analysis reveals that each model focuses on different aspects of the IT penetration process.*

REF: 29

(Dean 1974) Dean, Robert, C. Jr. "The Temporal Mismatch -Innovation's Pace vs Management's Time Horizon, " *Research Management* :12-15, May, 1974.

Category: innovation

Key words: technology management, research planning

Abstract/Summary: *The author discusses the negative impact on technology development of management's focus on short term gains. He provides comparisons between American view and those of our competitors.*

Referenced by (Przbylinski 1988)

(Dean 1987)   Dean, James W., Jr.,"Building the Future: The Justification Process for New Technology," in Johannes M. Pennings and Arend Buitendam (editors), *New Technology as Organizational Innovation: The Development and Diffusion of Microelectronics*, chapter 3, pages 35-58. Ballinger Publishing Company, Cambridge, MA, 1987.

Category: transition evaluation

Key Words: technology evaluation, capital budgeting, technology justification

Abstract/Summary: *This article summarizes a recent study by the author that looked at "innovation conceptualized as a decision making process", a concept proposed by Rogers and others. The sites for the study were five manufacturing organizations considering the adoption of advanced manufacturing technologies, such as computer-aided design or manufacturing requirements planning. Data came from both semi-structured interviews and archival materials, e.g., .internal memos, letters to and from vendors, etc. Dean used Downs and Mohr's "decision to innovate" as the unit of analysis. The study focuses on three components of the decision process: strategic/financial, social, and political. Each component is discussed in turn, with examples provided from the literature and the study itself. Each section includes strategies and tactics employed by individuals at different levels in the decision making process.*

Referenced by (Przbylinski 1988)

(Downs 1976) Downs, George W. and Lawrence B. Mohr, "Conceptual Issues in the Study of Innovation.," *Administrative Science Quarterly* 21:700-714, December, 1976.

Category: innovation

Key Words: innovation theory, innovation research

Abstract/Summary: *Downs and Mohr discuss four sources of instability in existing empirical research variation among primary attributes, interaction, ecological inferences and varying operationalizations of innovation. Based*

on this, they recommend seven characteristics *that new research should have to avoid these problems*

Referenced by (Przbylinski 1988)

(Downs 1979) Downs, George W. and Lawrence B. Mohr. Toward a Theory of Innovation. *Administration & Society* 10(4):379-408,February, 1979.

Category: innovation

Key Words: innovation research, innovation theory

Abstract/Summary: *This paper continues their work from 1976, defining new terminology for diffusion and adoption of innovations that is a first step in modeling the process* .

Referenced by (Przbylinski 1988)

(Dutton 1981) Dutton, William H., The "Rejection of an Innovation: The Political Environment of a Computer-Based Model," *Systems, Objectives, Solutions* 1(4): 179-202, 1981.

Category: innovation

Key Words: case study, innovation adoption

Abstract/Summary: *Dutton's paper provides a very detailed case study of the rejection of city planning model. It includes an in-depth analysis context, process and product characteristics.*

Referenced by (Przbylinski 1988)

(Elder 1986) Elder, Victoria., *Ada: A Case Study of Technology Transfer at DARPA.,* 1986

Category: technology transfer

Key Words: case study, DARPA

Abstract/Summary: *This paper was produced by the Center for the Productive Use of Technology at George Mason University. It continues the work started by Havelock and looks mainly at the context for Ada adoption in the Defense Advanced* Research Projects Agency.

Referenced by (Przbylinski 1988)

(Emerson 1983) Emerson, Thomas J., A. Frank Ackerman, Amy S. Ackerman, Priscilla Fowler,R. G. Ebenau and R. A. Rosenthal. "Training for Software Engineering Technology Transfer." in *IEEE Computer Society Workshop on Software Engineering Technology Transfer,* pages 34-41. IEEE, Silver Spring , MD. April, 1983.

Category: technology transfer

Key Words: consultative training, technical marketing

Abstract/Summary: *This paper describes the work of the SoftWare Engineering Technology Transfer group at AT +T Bell Laboratories, This group combined good technical marketing practices with highly tailored training in a process called consultative training that was very successful at transferring technologies into development groups at Bell Labs.*

Referenced by (Przbylinski 1988)

(Ettlie 1982) Ettlie, John E. and William P. Bridges "Environmental Uncertainty and Organizational Technology Policy," *IEEE Transactions on Engineering Management* EM-29 (1):2-10, February, 1982.

Category: innovation

Key Words: technology management

Abstract/Summary: *Ettlie and Bridges look at the impacts of an uncertain business environment on the adoption of process innovations* .

Referenced by (Przbylinski 1988)

(Ettlie 1987) Ettlie, John E. and William P. Bridges, "Technology Policy and Innovation in Organizations," in Johannes M. Pennings and Arend Buitendam (editors), *New Technology as Organizational Innovation: The Development and Diffusion of Microelectronics,* Chapter 6, pages 117-137. Ballinger Publishing Company, Cambridge, MA, 1987.

Category: innovation

Key Words: technology policy, technology strategy

Abstract/Summary: *This work continues the recent trend toward the view that organizational innovativeness and success are a function of technical strat*egy. Th*e authors contend that innovation* is *more likely in firms with an aggressive, forward-looking technology policy which they define as a "long range strat*egy *of the organization concerning the adoption of new process and material innovations and the origination of new product or service innovations." They employ* two *self-reporting research methods: self-administered .questionnaires and open-ended interviews. Their search revealed four key aspects of an aggressive firm's technology policy;* (1)*long-range commitment and investment in technological solutions to problems;(planning for the human resources needed to implement*

strategic technological plans, (3)openness to the environment with an eye toward tracking and forecasting technological trends ;and (4) structural adaptations such as unique positions, teams; task-forces, and mechanisms for functional integration in implement technology policies. One particularly interesting finding is that" although there are some industry differences, the greater the influence the government has as a factor in the firm's environment the less aggressive the firm's technology policy will be.:"

Referenced by (Przbylinski 1988)

(Farmer 1983) Farmer, J. Doyne, "The Dimension of Chaotic Attractors," *Physica* 7D, pp. 153-180, North-Holland Publishing Co., 1983.

REF: 46

(Feldman 1986) Feldman, Martha S., "Constraints on Communication and Electronic Mail.," in *Proceedings CSCW* ' 86, Pages 73-90. MCC Software Technology Program, Austin, TX, December, 1986.

Category: communication

Key Words: electronic mail, communication networks, weak ties

Abstract/Summary : *Feldman discusses how electronic media can create communication links between individuals who would otherwise not share information.. Granovetter's work on weak ties suggests that these new interactions may greatly influence behavior in the organization in question.*

Referenced by (Przbylinski 1988)

(Fichman 1993) Fichman, Robert G. and Kemerer, Chris F., "Adoption of Software Engineering Process Innovations: The Case of Object Orientation", *Sloan Management Review,* Winter (1993) 7-22

Category: Technology Introduction

Key Words: process innovations, object orientation, 4GL, RDB, Diffusion of Technology, Economics of Technology Standards, relative advantage, compatibility, complexity, trialability, observability, prior technology drag, irreversibility of investments, sponsorship, expectations

Abstract/Summary:

Software Development, unlike hardware development, seems to be plagued with constant problems. This stems from the fact that Software Engineering is still relatively new and undeveloped. This paper analyzes the adoption

of three technologies: "structured methodologies," fourth-generation programming languages (4GLs), and relational databases management systems (RDBs). The analysis of these technologies is from two perspectives: from the Diffusion of Technology (DOI) perspective, and from the Economics of Technology Standards perspective.

This paper then goes on to discuss the adoption of Object-Orientated (OO) Software Engineering Process Technologies. First, it gives an overview giving an overview of the concepts of OO. Then, based on the analysis of the older technologies, the authors predict that OO technology will not be quickly adopted outside of academia.

REF: 22

(Fichman 1994) Fichman-RG; Kemerera-CF, "Toward A Theory Of The Adoption And Diffusion Of Software Process Innovations," in *Diffusion, Transfer and Implementation of Information Technology*, in Levine, Linda, ed., proceedings of the IFIP TC8 Working Conference on Diffusion, Transfer and Implementation of Information Technology, Software Engineering Institute, Carnegie Mellon Institute, Pittsburgh, PA, North Holland, Amsterdam,, London, New York, Tokyo, 1994.

SOURCE: IFIP-Transactions-A-(Computer-Science-and-Technology). vol.A-45; 1994; p.23 –30

ABSTRACT: *It has become increasingly clear that no single, strongly predictive theory of innovation adoption and diffusion is likely to emerge. One response to this problem is to work at a higher level of abstraction and to identify general classes of explanatory factors or characteristic patterns related to adoption and diffusion of broadly defined innovations in broadly defined contexts. Another response is to narrow the focus to more specific innovations and contexts, and to develop a more strongly predictive theory centered around the distinctive characteristics of those innovations and contexts. This paper, takes the latter approach, and, in particular, argues that software process innovations (SPIs) (defined as a change to an organization's process for producing software applications) are distinguished by two characteristics: strongly increasing returns to adoption and substantial knowledge barriers impeding adoption. The combination of these two factors suggests that the study of the adoption and diffusion of SPIs*

across the internal IS units of large organizations will require new explanatory variables and knowledge of new patterns of diffusion.

REF: 22

(Fowler 1993) Fowler, Priscilla and Levine, Linda, "Conceptual Framework for Software Technology Transition," Technical Report, CMU/SEI-93-TR-31, ESC-TR93-317, December 1993.

ABSTRACT: *A conceptual framework that integrates and describes the intersections of three life cycles of software technology transition: research and development, new product development, and adoption and implementation in organizations. We then apply the framework to the technology transition experiences of the Software Engineering Institute.*

REF: 38

(Fowler 1994) Fowler P. and L. Levine, "From theory to practice: Technology Transition at the SEI", 1060-3425/94 1994 IEEE, p 483-497

Category: Technology Diffusion

Key Words: Diffusion models, diffusion process, technology management, mobile phones

Abstract/Summary: *There are 3 life cycles of technology transition: research and development, new product development, and implementation. This paper discusses the need for common terms in comparing development of products and the life cycles in depth.*

REF: 32

(Freeman 1988) Freeman, Peter, "Transfer bridge for software technology", TH0218-8/88 IEEE p 8-12

Category: Software Technology

Key Words: transfer process, application, software technology, transfer bridge

Abstract/Summary: *This paper talks about a proposed idea to span the gap between production and application of software technology. The first model of technology transfer consisted of three major functions: creation, transfer, and application of software. The transfer bridge will have realistic educational settings for professionals in post-graduate training. The three major implementation concerns are cooperation, stability, and complimentarity. Technology transfer problems must be attacked with more*

than one solution. The "transfer bridge" concept is just one of many ideas.

REF: 3

(Froehling 1981) Froehling, Harold, Crutchfield, J.P., Farmer, Doyne, Packard, N.H. and Shaw, Rob, ""n Determining the Dimension of Chaotic Flows," *Physica* 3D, pp. 605-617, North Holland Publishing Co., 1981.

REF: 31

(Gerhart 1994) Gerhart, S..L, "The MCC Formal Methods Transition Study: Technology Transfer For Complex Information Technology And Processes," in Levine, Linda, ed., proceedings of the IFIP TC8 Working Conference on Diffusion, Transfer and Implementation of Information Technology, Software Engineering Institute, Carnegie Mellon Institute, Pittsburgh, PA, North Holland, Amsterdam, London, New York, Tokyo, 1994.

SOURCE: IFIP-Transactions-A-(Computer-Science-and-Technology). vol.A-45; 1994; p.249-55

ABSTRACT: *This paper describes a technology transfer model used at MCC, a research consortium in Austin Texas, in 1990-1991. It discusses the purpose of the project. It looks at the nature of the model. It gives some details of the project. It discusses experience before, during, and after the project and makes some generalizations. The interesting features of the project from a technology transfer perspective are: a successfully executed project, but with inconclusive results due to its untimely demise; a combination of scholarly investigation, ambitious experimentation, and practical, user-oriented delivery; a very broad, large scale exploration of a complex subject area, driven by templates and assessment criteria; and an example of what can be produced and a process that works in a short (one year) time frame.*

REF: 8

(Gerstenfeld 1983) Gerstenfeld, Arthur and Paul De. Berger,."From Basic Research to Application, A Model of Effective Technology Transfer." Technical Report TR-ONR-2, Office of Naval Research, August 1983.

Category: innovation

Key Words: technology transfer, technology management

Abstract/Summary: *This study traced sixty projects with links between research and*

application through inter views with over 100 individual engineers and scientists. The authors propose a linear model .of technology transfer which includes organizational ,environmental, people and resources issues, each a composite of a number of factors .Included are many illustrations of successful from transfer mechanisms used by the firms interviewed. Their research showed that the most important factors were management attitude, entrepreueurship, timing and dollars. The report concludes with a number of research questions, including the authors proposal that firms use "research portfolios", with the normal financial risks factors replaced by "probabilities of application" and "estimated time-of application.

Referenced by (Przbylinski 1988)

(Ginn 1994) Ginn, M.L., "The Transitionist As Expert Consultant: A Case Study Of The Installation Of A Real-Time Scheduling System In An Aerospace Factory" in Levine, Linda, ed., *Proceedings of the IFIP TC8 Working Conference on Diffusion, Transfer and Implementation of Information Technology,* Software Engineering Institute, Carnegie Mellon Institute, Pittsburgh, PA, North Holland, Amsterdam,, London, New York, Tokyo, 1994.

SOURCE: IFIP-Transactions-A-(Computer-Science-and-Technology). vol.A-45; 1994; p.179-98

ABSTRACT: *A small group of transitionists implemented a computerized scheduling system in an aerospace factory. The author, one of these transitionists, uses a qualitative analysis to identify the dynamics that prevented full and rapid technology transition. This paper describes this project's diffusion process and compares and contrasts three cultures of inquiry (empirical-analytical, ethnography, and action research) appropriate to diffusion of innovation research. A new Four Hills model is introduced that can help assess risk and plan action steps in regard to four key roles: sponsors, transitionists, middle managers, and workers. The Four Hills model also can extend classic diffusion of innovation research, attributes of innovations. There is an important distinction between the implementation of the new system and the new work method, which is important in assessing implementation success. Finally, a metaphor may enhance understanding of a key implementation issue, middle managers acting as guardians of the social-work system which a new work method might disrupt.*

REF: 27

(Glass 1998) Glass, Robert L., "An Assessment of Systems and Software Engineering Scholars and Institutions (1193-1997," *The Journal of Systems and Software* 43, pp. 59-64, 1998.

REF: 8

(Glasson 1994) Glasson,B.C., ISTRAD: "Toward A National Information Systems And Technology Research And Development Association," in Levine, Linda, ed., proceedings of the IFIP TC8 Working Conference on Diffusion, Transfer and Implementation of Information Technology, Software Engineering Institute, Carnegie Mellon Institute, Pittsburgh, PA, North Holland, Amsterdam,, London, New York, Tokyo, 1994.

SOURCE: IFIP-Transactions-A-(Computer-Science-and-Technology). vol.A-45; 1994; p.333-44

ABSTRACT: *The International Federation for Information Processors (IFIP) aims to foster research, development, application, education and information dissemination in all fields of informatics. IFIP works through a number of technical committees each focussing on one aspect of informatics. The technical committees in turn are responsible for a small number of working groups. Each working group focuses on some sub-set of the field covered by its parent technical committee. Given its objectives, IFIP is uniquely placed to foster both hard and soft technology diffusion, it is doing so with mixed success. The author describes how the competitive advantage offered by one IFIP technical committee has been used to improve technology diffusion in the field of information systems nationally and internationally. He describes a series of activities aimed at building up an information systems and technology research community in Australia. An initial, key activity, was to run an IFIP supported national seminar series on the "State of the Art in Information Systems". The seminar series was used to launch the concept of a national information systems and technology research and development association (ISTRAD). An additional outcome was a "State of the Art in Information Systems" video which is being distributed world-wide as an educational resource.*

REF: 12

(Granovetter 1973) Granovetter, Mark S. "The Strength of Weak Ties," *American Journal of Sociology* 78(6), pp. 1360-1380, 1973.

Category: communication

Key Words: communication networks, influence networks

Abstract/Summary: *This paper discusses weak ties, a concept that de-scribes how individuals who are weakly linked in social terms can exert substantial influence in communication networks. Weak ties have implications for technology dissemination activities.*

Referenced by (Przbylinski 1988)

(Gross 1984) Gross, Pamela H.B. and Michael J. Ginzberg "Barriers to the Adoption of Application Packages," *Systems, Objectives, Solutions* 4: pp. 211-226, 1984.

Category: innovation

Key Words: innovation adoption

Abstract/Summary: *This paper describes a qualitative study of technology adoption. It includes lengthy lists of factors to consider during technology insertion.*

Referenced by (Przbylinski 1988)

(Grossman 1974) Grossman, Lee, *The Change Agent,* AMACOM, New York, 1974.

Category: organization change

Key Words: change agent

Abstract/Summary: *This book now out of print, takes an anecdotal approach to describing the roles and responsibilities of change agents in organizations.*

Referenced by (Przbylinski 1988)

(Gruber 1969a) Gruber, William H., and Donald G. Marquis, *Factors in the Transfer of Technology*, Massachusetts Institute of Technology, Cambridge, MA, 1969.

Category: technology transfer

Key Words: technology transfer, innovation

Abstract/Summary: *This book is basically a workshop proceedings from a large workshop lead at MIT attended by some of the leaders in the field. Many papers in the book are referenced separately in this list. The summary paper written by Gruber and Marquis is outstanding..*

Referenced by (Przbylinski 1988)

(Gruber 1969b) Gruber, William H. and Donald G. Marquis, "Research on the Human Factor in the Transfer of Technology," in *Factors in the Transfer of Technology,* Pages 255-282. Massachusetts Institute of Technology, Cambridge, MA, 1969.

Category: technology transfer

Key Words: innovation

Abstract/Summary: *This summary paper contains sections on the following determinants of technology transfer ;training and experience; individual personality characteristics; communication patterns; organizational effects; mission orientation; and motivation.*

Referenced by (Przbylinski 1988)

(Havelock 1985) Havelock, Ronald G. and David S. Bushnell. 'Technology Transfer at DARPA -The Defense Advanced Research Projects Agency: A Diagnostic Analysis." Technical Report DTIC AD-A164 457,Technology Transfer Study Center, George Mason University, December, 1985.

Category: technology transfer

Key Words: technology transfer, DARPA

Abstract/Summary: *This paper provides an in-depth look at how technology transfer is planned as a multi-stage process at DARPA. It discusses the problems inherent in trying to get government, defense, academic and contractors to cooperate to a common end. The case study by Elder is a continuation of this work.*

Referenced by (Przbylinski 1988)

(Heidtman 1994) Heidtman, S.E., "Exploration Of An Incremental Approach To Technology Transfer And The Issues Affecting Its Implementation," in Levine, Linda, ed., proceedings of the IFIP TC8 Working Conference on Diffusion, Transfer and Implementation of Information Technology, Software Engineering Institute, Carnegie Mellon Institute, Pittsburgh, PA, North Holland, Amsterdam, London, New York, Tokyo, 1994.

SOURCE: IFIP-Transactions-A-(Computer-Science-and-Technology). vol.A-45; 1994; p.347-51

ABSTRACT: New system development technologies promise higher quality, less costly systems. However, implementing new technologies is a difficult, often unsuccessful task. Some of the problems

associated with transition may be resolved through an incremental technology transition process. However, the benefits of using this model are, at this time, unproven. In order to justify the effort required to implement the model, its benefits should be clearly established. The author offers a summary of the incremental model and discusses barriers to validation and implementation.

REF: 0

(Hook 1986) Hook, Audrey *A.,* Terry Mayfield, Thomas Frazier, Alan K. Graham and David Kreutzer, "Cost Effectiveness Tradeoffs in Computer Standardization and Technology Insertion," Technical Report P-1931, Institute for Defense Analysis, June, 1986.

Category: transition evaluation

Key Words: decision support system,cost model

Abstract/Summary: *This report discusses the feasibility of developing a decision support system to aid in the use of software standards and in the development of strategies for technology insertion. In this study performed for the Ada Joint Program Office, IDA developed a prototype system which could simulate some effects of standardization policies on related technologies and Mission Critical Computer Resources costs. The preliminary result obtained by their prototype 'is that standardization policies have a payoff two to three orders of magnitude greater than their costs.*

Referenced by (Przbylinski 1988)

(Hornbach 1988) Hornbach, Katherine, "The Role of Support Staff in the Successful Introduction of New Tool Technology", TH0218-8/88 IEEE pp. 74-77, 1988.

Category: Technology introduction, Technology management

Key Words: Administrative support, Process support, Cultural issues

Abstract/Summary: This paper discusses the responsibilities needed in order to use new tool technology successfully. It provides a detailed list of the tasks needed to be performed by a support person and includes a real-life example illustrating the role of a support person in tool introduction. The paper focuses on the role of a support person in both administrative and process support tasks. It describes the necessity of the support person when dealing with new tool technology.

REF: 3

(Huber 1991) Huber, George P. "Organizational Learning:: The Contributing Processes and the Literatures", Organization Science, Vol 2, No. 1, February 1991.

REF: 204

(Humphrey 1987a) Humphrey, Watts S. and William Sweet. "A Method for Assessing the Software Capability of Contractors*,"* Preliminary Report CMU/SEI-87 TR-23, Software Engineering Institute, July, 1987.

Category: organizational change

Key Words: process assessment, process consultation

Abstract/Summary: *This report contains a preliminary version of an assessment instrument jointly developed by the SEI and Mitre for the Air Force. It allows contractors to perform self-assessments of their software capabilities to pinpoint areas for possible improvement. If properly used, this tool can help determine technology for insertion.*

Referenced by (Przbylinski 1988)

(Humphrey 1987b) Humphrey, Watts S. 'Characterizing the Software Process: A Maturity Framework," Technical Report DTIC ADA 1182895, Software Engineering Institute, June 1987

Category: organizational change

Key Words: process assessment, process consultation

Abstract/Summary: *This paper provides the foundation for the process improvement Work at the SEI. It describes a five stage framework for the maturity of an organization's software development activities based on Humphrey's work at IBM.*

Referenced by (Przbylinski 1988)

(Huseth 1988) Huseth, Steve, "The Cost of Technology Transfer," THO218-8/88, IEEE, pp. 80-81, 1988.

REF: 5

(IEEE 83a) The Institute of Electrical and Electronics Engineers, Inc. *IEEE Standard Glossary of Software Engineering Terminology.* The Institute of Electrical and Electronics Engineers, Inc., New York, NY, 1983.

Referenced by (Przbylinski 1988)

(IEEE 83b) IEEE Computer Society, Konover Hotel, Miami Beach, Florida. IEEE Computer Society Workshop on Software

Engineering Technology Transfer, April 25-27, 1983.

Category: technology transfer

Key Words: technology transfer

Abstract/Summary: *This proceedings describes the first workshop of this kind held to consider software issues. While many of the papers are good, the best outputs here are the panel summaries included in the front of the proceedings. Unfortunately, the panels recommendations were not .followed up by the following workshops.*

Referenced by (Przbylinski 1988)

(IEEE Std 1348-1995) IEEE Std 1348-1995, IEEE Recommended Practice for the Adoption of Computer-Aided Software Engineering (CASE) Tools, ISBN 1-55937-591-4, IEEE, 1996.

REF: 25

(Ignace 1994) Ignace, S.J; Sedlmeyer, R.L.; Thuente, D.J., "Integrating Rate-Monotonic Analysis Into Real-Time Software Development," in Levine, Linda, ed., proceedings of the IFIP TC8 Working Conference on Diffusion, Transfer and Implementation of Information Technology, Software Engineering Institute, Carnegie Mellon Institute, Pittsburgh, PA, North Holland, Amsterdam,, London, New York, Tokyo, 1994.

SOURCE: IFIP-Transactions-A-(Computer-Science-and-Technology). vol.A-45; 1994; p.257-74

ABSTRACT: *Rate-monotonic analysis (RMA) is a new technology that provides an engineering basis for designing real-time systems. During the last two years we have made significant progress in integrating this technology with our standard software development process. We give an account of our activities in procuring expertise in, promoting the use of, and providing training for rate-monotonic analysis. We present our model for technology acquisition and discuss how our experiences relate to established models of technology transfer. We also detail two case studies which served as convincing examples of the utility of this technology.*

REF: 14

(Isenson 1969) Isenson, Raymond S. Project Hindsight: "An Empirical Study of the Sources of Ideas Utilized in Operational Weapon Systems," in William H.Gruber and Donald G. Marquis (editors), *Factors in the Transfer of Technology,* chapter 10, pages 155-176. The M.I.T. Press, Cambridge, MA, 1969.

Category: innovation

Key Words: technology development

Abstract/Summary: *There are implicit assumptions made by various government agencies that their research and development money is well spent, with results flowing into systems into production. This study considered just that question. The author concludes, among other things, that this may be true, although there may be a time lag of up to ten years*

Referenced by (Przbylinski 1988)

(Jaakkola 1995) Jaakkola, Hannu, :Comparison and Analysis of Diffusion Models", p 65-82, 1995.

Category: Technology Diffusion

Key Words: Diffusion models, diffusion process, technology management, mobile phones

Abstract/Summary: *A real diffusion process is too complex to put into a model accurately. We try our best to model what we see but we cannot understand all of the interrelations between variables. There are several types of models, each with their own attributes. This paper focuses on which models best fit each situation.*

REF: 33

(Jeffrey 1988) Jeffrey, H Joel, " A Unifying Comprehensive Framework for Software Technology Transfer", TH0218-8/88 p82-85

Category: Technology transfer

Key Words: Conceptual tools, Linguistic tools, Methodological tools, Descriptive psychology, Pragmatic evaluation, Communities, Sociology, Action

Abstract/Summary:

This paper describes the usage of Descriptive Psychology to simplify the process of transferring technology to more communities. It explains the human actions needed to transfer technology successfully, describing the suitable formulation that is critical in explaining the key differences between descriptive psychology and other approaches. It also describes the formulation that allows successful communication. The paper provides a parametric analysis of human behavior and communities while listing the steps needed to

gain cooperation in a project. Lastly, the paper contains a pragmatic evaluation from the applications of these formulations through Putnam.

REF: 9

(Kamm 1986) Kamm, Judith B. "The Portfolio Approach to Divisional Innovation Strategy, " *Journal of Business Strategy* 7 (1): pp. 25-36, Summer, 1986

Category: innovation

Key Words: management of innovation, strategic planning

Abstract/Summary: *Organizations often must juggle the varied needs for product, process and administrative innovation. In addition, failure in one type of innovation can lead to failure in the others. The author proposes that management (on the divisional level) use a portfolio approach to balance these needs. Innovation projects are evaluated on two different scales: form and objectives. Form consists of product, process and administrative innovation, classified by whether the changes are "revolutionary" or "evolutionary". The objectives included are maintaining the business, expanding the business or using capacity, classified this time into short and long-term categories. The author goes on to discuss the problems that can arise using this approach, and gives examples from field work from semiconductor and pharmaceutical firms.*

Referenced by (Przbylinski 1988)

(Kanter 1983) Kanter, Rosabeth Moss, "Change Masters And The Intricate Architecture Of Corporate Culture Change," *Sloan Management Review*, pp.18-28, October, 1983.

Category: organizational change

Key Words: change agents, innovation, innovation roles

Abstract/Summary: *This paper is an excerpt from her book The Change Masters - case studies of change in high tech organizations,*

Referenced by (Przbylinski 1988)

(Kappelman 1994) Kappelman-LA; McLean-ER, " User engagement in information system development, implementation, and use: toward conceptual clarity," in Levine, Linda, ed., proceedings of the IFIP TC8 Working Conference on Diffusion, Transfer and Implementation of Information Technology, Software Engineering Institute, Carnegie Mellon

Institute, Pittsburgh, PA, North Holland, Amsterdam,, London, New York, Tokyo, 1994.

SOURCE: IFIP-Transactions-A- (Computer-Science-and-Technology). vol.A-45; 1994; p.199-214

ABSTRACT: *Although a great deal of research attention has been given to the roles of users in information system development and implementation, there is a scarcity of common models and measurements. Moreover, the empirical evidence regarding the value of such user roles is mixed. As a consequence, it is difficult to make comparisons and generalizations based upon this literature. This state of affairs is the result of the varied conceptualizations and operationalizations of the constructs utilized, the somewhat ambiguous use of terminology, and other methodological deficiencies. This paper presents a more consistent vocabulary to be used with regard to the various ways in which users can be engaged in the processes of information system development, implementation, and use. Drawing upon recent information systems studies, as well as the psychological, consumer, and organizational behavior literature, a taxonomy for the engagement of users with information systems is proposed. This framework recognizes distinctions among the psychological and behavioral components, as well as the task and product objects of such engagements. Preliminary evidence suggests that such distinctions can improve the research that is currently being undertaken in this important area.*

REF: 98

(Kautz 1994) Kautz K. ; McMaster T., "The failure to introduce system development methods: a factor-based analysis," in Levine, Linda, ed., proceedings of the IFIP TC8 Working Conference on Diffusion, Transfer and Implementation of Information Technology, Software Engineering Institute, Carnegie Mellon Institute, Pittsburgh, PA, North Holland, Amsterdam,, London, New York, Tokyo, 1994.

SOURCE: IFIP-Transactions-A- (Computer-Science-and-Technology). vol.A-45; 1994; p.275-87

ABSTRACT: *Structured methods for the development of computer-based systems have been promoted for more than 20 years, but still they are not used in many organisations. We investigate the issue of failed attempts to*

implement structured methods. On the basis of a literature study we present a framework for analyzing failure and introduce a case study showing how such a failure occurred in a practical situation. Through critical examination of a number of factors we formulate some recommendations. These are neither generalizable nor offer a guaranteed prescription for success, but we feel that they have some value in that they may help to minimise the risk of failure for the future introduction of structured development methods.
REF: 24

(Klempa 1994) Klempa-MJ, "Management of information technology diffusion: a meta-force integrative contingency diffusion model," in Levine, Linda, ed., proceedings of the IFIP TC8 Working Conference on Diffusion, Transfer and Implementation of Information Technology, Software Engineering Institute, Carnegie Mellon Institute, Pittsburgh, PA, North Holland, Amsterdam,, London, New York, Tokyo, 1994.
SOURCE: IFIP-Transactions-A-(Computer-Science-and-Technology). vol.A-45; 1994; p.31 –52

ABSTRACT: *Prior research analyzes diffusion of information technology (IT) from disparate theoretical frameworks, often cross sectional in nature, and not utilizing interactionist perspectives. This paper proposes an original, holistic, two-tiered contingency IT diffusion model. The first tier identifies three meta-forces which drive information technology acquisition and diffusion (IT/AD)-organization culture, organization learning and knowledge sharing. Both the characteristics of, as well as the interaction of, these three meta-forces determines the organization's creativity, synergy, and leveraging of IT/AD. These three meta-forces interact recursively, expressed via both rational and political organization processes. Unlike previous nominal IT/AD diffusion models, the IT/AD contingency model proposed herein is parsimonious. The second tier of the IT/AD model delineates secondary IT/AD forces (moderating variables) which further enhance or inhibit IT/AD. The second tier also considers the decision-making/diffusion process coupling. The complete IT/AD contingency model hypothesizes clusters of S-shaped diffusion curves. Future research directions, utilizing positivist, interpretive, and combined positivist/interpretive*

venues, as suggested by the model, are presented.
REF: 113

(Kuvaja 1994) Kuvaja, P. "Productivity of CASE technology implementation in SW development and maintenance on the third maturity level," in Levine, Linda, ed., proceedings of the IFIP TC8 Working Conference on Diffusion, Transfer and Implementation of Information Technology, Software Engineering Institute, Carnegie Mellon Institute, Pittsburgh, PA, North Holland, Amsterdam,, London, New York, Tokyo, 1994.
SOURCE: IFIP-Transactions-A-(Computer-Science-and-Technology). vol.A-45; 1994; p.215-29.

ABSTRACT: *This paper reports the effects of CASE technology implementation on the productivity of software processes at the third (defined) maturity level. The results were gathered in a life-cycle simulation in which 11 lower and upper CASE technologies were used to develop and maintain the same test software system. Productivity was measured in labour hours spent on one function point. The results show differences and similarities in productivity between three classes of CASE technology and between development and maintenance.*
REF: 47

(Leon 1994) Leon-G; Carracedo-J; Yelmo-JC; Sanchez-C; Moreno-JC; Gil-JJ; Carrasco-J., "An industrial experience of using an incremental model of technology transfer of formal development methods," in Levine, Linda, ed., proceedings of the IFIP TC8 Working Conference on Diffusion, Transfer and Implementation of Information Technology, Software Engineering Institute, Carnegie Mellon Institute, Pittsburgh, PA, North Holland, Amsterdam,, London, New York, Tokyo, 1994.
SOURCE: IFIP-Transactions-A-(Computer-Science-and-Technology). vol.A-45; 1994; p.289-308

ABSTRACT: *This paper describes the process of transferring formal methods to the industry and specifically LOTOS and SDL as representative Formal Description Techniques (FDTs). From this purpose, a technology transfer model is described in order to accelerate their use. This model is conceptually presented under an incremental approach where the transference is done in several phases (or*

cycles). The first cycle is termed academic; there, the formalism and its theoretical framework is introduced. The second one is the methodological cycle where the emphasis is placed on the design of large specifications and its evaluation in a specific application domain to derive a sound methodological basis. The industrialization cycle considers the problems of introduction of the selected technology in the industrial practice under specific constraints. The experience of using this model in one research project (MEDAS) is outlined. The project included the development of three large case studies in the telecom field. From this experience a set of recommendations about how to transfer FDTs based on the characterization of industries w.r.t. software technology factors is proposed.

REF: 16

(Leonard-Barton 1985a) Leonard-Barton, Dorothy "Experts as Negative Opinion Leaders in the Diffusion of a Technological Innovation," *Journal of Consumer Research* 110: pp. 914-926, March, 1985.

Category: innovation

Key Words: diffusion research, transition barriers

Abstract/Summary: *Much diffusion of innovation research suffers from a "pro-innovation" bias, that is, studies look at the positive aspects and forces in the spread of an innovation. In this case, the author is interested in "negative" opinion leaders, individuals with stature in a given field that oppose adoption of a given innovation. Leonard-Barton conducted a study of the diffusion of the use of non-precious alloys by prothodontists (dentists who specialize in crowns and bridges as restorations). While most researchers take a sociometric approach aimed at discovering direct verbal communication patterns within a closed community, this study used a lengthy questionnaire administered to two populations: a sample from the greater Boston area and a national sample obtained from professional societies. While many of her hypotheses were rejected there were some interesting results Positive opinion leaders must propagate new skills in addition to providing information. Negative opinion leaders need only denigrate the innovation. Leonard Barton postulates that this is true any time the innovation requires acquisition of complex skills in addition to those required for the alternative product or method. Equally important is the finding that opinions*

formed on the basis of information alone are just as negative as those based on personal experience with the innovation.

Referenced by (Przbylinski 1988)

(Leonard-Barton 1985b) Leonard-Barton, Dorothy and William A. Kraus, "Implementing New Technology,".*Harvard Business Review, pp.*102-110, November-December, 1985.

Category: organization change

Key Words: innovation, innovation roles, risk reduction

Abstract/Summary: *Leonard-Barton discusses roles in the innovation process, the use of pilot projects and other general risk reduction strategies .*

Referenced by (Przbylinski 1988)

(Lien 1994) Lien-L, "Transferring technologies from developed to developing industrial and commercial environments," in Levine, Linda, ed., proceedings of the IFIP TC8 Working Conference on Diffusion, Transfer and Implementation of Information Technology, Software Engineering Institute, Carnegie Mellon Institute, Pittsburgh, PA, North Holland, Amsterdam,, London, New York, Tokyo, 1994.

SOURCE: IFIP-Transactions-A-(Computer-Science-and-Technology). vol.A-45; 1994; p.87 -98

ABSTRACT: *The author presents the current practice of training and operations to increase the probability of successful technology and information transfer. He addresses the process, content, management and factors that affect transfer. He discusses the influence of the project dynamic, capacity of suppliers of technology to transfer, and receivers to accept and apply. The author offers a management framework that allows for effective definition, control and verification that technology has been transferred. He presents a mathematical model that addresses eight factors influencing transfer, that can be effectively used to predict the probability of successful transfer, and as a method to develop alternative transfer scenarios.*

REF: 6

(Lindgaard 1994) Lindgaard-G, "Some important factors for successful technology transfer", in Levine, Linda, ed., proceedings of the IFIP TC8 Working Conference on Diffusion, Transfer and Implementation of Information Technology, Software Engineering Institute,

Carnegie Mellon Institute, Pittsburgh, PA, North Holland, Amsterdam,, London, New York, Tokyo, 1994.

ABSTRACT: *This paper discusses some important factors which, to a great extent, determine the success of technology-based products, services and features in the market place. In particular, it addresses the issues of usefulness, usability and implementation strategies employed by organizations undergoing technological changes. It is shown that usefulness, or the degree to which products match users' needs, can determine the success or failure of certain products, and that, in many cases, the number of smart features available to users by far outweigh those that are actually being used. Three studies are discussed to support this point. It shows further that usability is quantifiable and measurable, and that product development should be guided by usability goals and criteria. Usability can and should be evaluated throughout the development process in an iterative fashion to avoid usability disasters at the last minute before a product is released. Successful transfer of technology, it is argued, is related to careful strategic planning and involvement of people whose jobs will be affected by the introduction of new technology.*
REF: 36

(Lopata 1994) Lopata-CL, "Implementation scripts: a new approach to modeling the process," in Levine, Linda, ed., proceedings of the IFIP TC8 Working Conference on Diffusion, Transfer and Implementation of Information Technology, Software Engineering Institute, Carnegie Mellon Institute, Pittsburgh, PA, North Holland, Amsterdam,, London, New York, Tokyo, 1994.

ABSTRACT: *This paper presents a new, empirically grounded, process model of the implementation of information technology in an organization. The model is based on a longitudinal investigation of the implementation of a computer -based information management system in a three-college library consortium. Data were collected through interviews with, and observations of, participants in the implementation process at various stages in that process, and through an analysis of pertinent documents produced by the organization. Implementation is conceptualized here as a process of mutual adaptation: both the technology and the organization, where that technology was implemented, were adapted as the process unfolded. Using events analysis, in combination with script theory, instances of adaptation are presented in the context of other organizational events and interruptions to the process. Patterns of events are then identified and aggregated to form scripts for the different periods of the implementation process.*
REF: 7

(Maidique 1980) Maidique, Modesto A, "Entrepreneurs, Champions, and Technological Innovation," *Sloan Management Review* 21(2), pp.5976, Winter, 1980.
Category: innovation
Key Words: innovation roles, risk reduction
Abstract/Summary: *Maidique summarizes much of the existing work on roles in the innovation process .*
Referenced by(Przbylinski 1988)

(Mankin 1984) Mankin, Don, Tora K. Bikson and Barbara Gutek. "Factors in Successful Implementation of Computer-Based Office Information Systems: A Review of the Literature with Suggestions for OBM Research," *Journal of Organization Behavior Management* (6/3/4): pp. 1-20, Fall/Winter, 1984.
Category: innovation
Key Words: technology transfer, innovation, innovation roles
Abstract/Summary: *This paper starts with a good, short review of the literature on innovation acceptance. It continues to develop a communication motivation oriented model of organizational interaction.*
Referenced by(Przbylinski 1988)

(Mitchell) Mitchell, K. I., "Technology Transfer to & from the Industrial Sector," 495-496.
Category: Technology Transfer
Key Words: communication, cooperation
Abstract/Summary: *This paper provides an idea of what it takes to transfer technology to and from the Industrial Sector. Some of the ideas discussed include communication, strategic alliances, the forms of strategic alliance, the three stage model,*

funding, personnel exchange, artificial barriers, transfusion through graduates, and diffusion models.

REF:

(Montealegre 1994) Montealegre-R; Applegate-LM, "Information technology and organization change: lessons from a less - developed country," in Levine, Linda, ed., proceedings of the IFIP TC8 Working Conference on Diffusion, Transfer and Implementation of Information Technology, Software Engineering Institute, Carnegie Mellon Institute, Pittsburgh, PA, North Holland, Amsterdam,, London, New York, Tokyo, 1994.

SOURCE: IFIP-Transactions-A- (Computer-Science-and-Technology). vol.A-45; 1994; p.99 -131

ABSTRACT: *The introduction and assimilation of technology within organizations has been viewed as a process of organizational change that involves the mutual adaptation of environment, organization, individual/work group, and information technology. The authors present a conceptual framework for studying this complex phenomena and illustrate the use of the framework by analyzing the introduction of information technology within a Guatemalan sugar company.*

REF: 39

(Myers 1985) Myers, Ware MCC: "Planning the Revolution in Software," I*EEE Software* 2(6), pp. 68-73, November, 1985.

Category: technology transfer

Key Words: research consortia

Abstract/Summary: *This interview with Les Belady provides insight into MCCs approach to technology transfer (in*1985).

Referenced by (Przbylinski 1988)

(Paulish 1994) Paulish D.J., "Experience with software measurement technology transfer," in Levine, Linda, ed., proceedings of the IFIP TC8 Working Conference on Diffusion, Transfer and Implementation of Information Technology, Software Engineering Institute, Carnegie Mellon Institute, Pittsburgh, PA, North Holland, Amsterdam,, London, New York, Tokyo, 1994.

SOURCE: IFIP-Transactions-A- (Computer-Science-and-Technology). vol.A-45; 1994; p.381-94

ABSTRACT: *The author describes the experience of Siemens AG in transferring technology associated with the application of software measurement to software development organizations. This experience was obtained as part of the Consortium working on the ESPRIT II PYRAMID Project. The author describes some of the methods used for technology transfer. He summarizes the lessons learned about technology transfer as a result of the project. An approach is given to measure technology transfer exposure, and the Siemens results for the PYRAMID Project are given. The benefits to Siemens resulting from exploitation of PYRAMID Project results are summarized.*

REF: 8

(Pennings 1987) Netherlands Institute for Advanced Studies, Groningen, The Netherlands. *New Technology as Organizational Innovation: The Development and Diffusion of Microelectronics,* 1987.

Category: innovation

Key Words: innovation, organizational change, technology selection, technology justification.

Abstract/Summary: *This book, published in* 1987, is *a collection of related articles on innovation, primarily in high-tech industries. Its thirteen chapters each discuss a different topic, including technology justification, technology policy, high technology marketing, and the impacts of information technology.*

Referenced by (Przbylinski 1988)

(Pfleeger 1999) Pfleeger, S. L., "Understanding and improving technology transfer in software engineering", The Journal of Systems and Software 47 (1999) 111-124

Category: Technology Introduction

Key Words: technology transfer

Abstract: Even at its quickest, it usually takes decades for a new technology to be widely adopted as standard practice in government and industry. This paper, though it never explicitly defines "technology transfer," describes the processes in which technology is transferred from idea ("Technology Creation") to adoption ("Technology Diffusion"). It describes the processes and roles involved. This paper also describes ways in which the speed of technology transfer can be increased.

REF: 15

(Popham 1975) Popham, W. James. *Educational Evaluation,* Prentice-Hall, Inc., Englewood Cliffs, NJ, 1975.

Category: transition evaluation
Key Words: evaluation models
Abstract/Summary: *This book contains models and methods for educational evaluation that may be applicable to technology transfer.*

Referenced by (Przbylinski 1988)

(Pries 1994) Pries-Heje-J; Lauesen-S; Schroder-B, "Barriers to software technology transfer in the Danish electronic equipment industry," in Levine, Linda, ed., proceedings of the IFIP TC8 Working Conference on Diffusion, Transfer and Implementation of Information Technology, Software Engineering Institute, Carnegie Mellon Institute, Pittsburgh, PA, North Holland, Amsterdam,, London, New York, Tokyo, 1994.

SOURCE: IFIP-Transactions-A-(Computer-Science-and-Technology). vol.A-45; 1994; p.133-7

ABSTRACT: *The authors summarize a study of software technology transfer from academia to the Danish electronic equipment industry. The study revealed that only very few research results are transferred. The lack of technology transfer is caused by researchers' lack of knowledge of the real problems in industry. The study also showed that even very good and very relevant results sometimes failed to be taken into regular use in the industry. Many different barriers cause this failure. Finally the authors suggest how these large technology transfer problems could be overcome.*

REF: 5

(Quinn 1979) Quinn, James Brian., "Technological Innovation, Entrepreneurship, and Strategy," *Sloan Management Review* 20(3), pp. 19-30, 1979.

> **Category:** innovation
> **Key Words**: organizational evolution
> **Abstract/Summary**: *The author talks about idea generation and product development during the different stages of an organizations lifecycle. It includes discussion of the conflicts between corporate needs and entrepreneurship*
> Referenced by (Przbylinski 1988)

(Quinn 1982) Quinn, James Brian and James A. Mueller, "Transferring Research Results to Operations," in Michael L. Tushman and William L. Moore (editors), *Readings in the Management of Innovations,* pages 60-83. Ballinger Publishing Company, Cambridge, MA, 1982.

> **Category**: technology transfer
> **Key Words**: receptor groups, technology management
> **Abstract/Summary**: *In the authors opinion, certain management actions can stimulate the effective flow of technology within organizations. They describe a four-step program to achieve his end: examine resistances at critical technological points,. provide the information to target research toward company goals,. foster a positive motivational environment; and plan and control the exploitation of R&D results.*
> Referenced by (Przbylinski 1988)

(Raghavan 1986) Raghavan, Sidhar A. and Donald R. Chand "Diffusion of Software Engineering Methods*,"* Technical Report TR-86-10, Wang Institute of Graduate Studies, November, 1986.

> **Category**: innovation
> **Key Words**: innovation diffusion
> **Abstract/Summary:** *This technical report provides a good summary Everett Rogers' framework for diffusion of innovations.*
> Referenced by (Przbylinski 1988)

(Raghavan 1988) Raghavan, Sridhar, "Diffusion Software Engineering Innovation,", THO218-8/88 *IEEE*, pp. 116-118, 1988.

> REF: 1

(Raghavan 1989). Raghavan, Sridhar, and Chand, Donald R, "Diffusing Software-Engineering Methods", 0740-7459/89 IEEE, pp. 81-89

> **Category**: Diffusing Technology
> **Key Words**: transfer, practice, systematic understanding
> **Abstract/ Summary**: *Software Engineers are having a difficult time trying to find a good framework to study the nature of software-technology transfer. Although the field of software engineering has significantly grown, it has not changed the practice of software development. The problem of understanding software-technology transfer is the software-engineering innovators are trying to oversimplify or run away from the problems concerning technology transfer. To solve the overall problem it is helpful to get a very through understanding of the processes and problems and tackle the technology transfer problems head-on.*
> REF: 10

(Ramiller 1994) Ramiller, N.C.; Swanson, E.B., "Toward an institutional view of information technology diffusion, transfer, and implementation," in Levine, Linda, ed., proceedings of the IFIP TC8 Working Conference on Diffusion, Transfer and Implementation of Information Technology, Software Engineering Institute, Carnegie Mellon Institute, Pittsburgh, PA, North Holland, Amsterdam, London, New York, Tokyo, 1994.

> SOURCE: IFIP-Transactions-A-(Computer-Science-and-Technology). vol.A-45; 1994; p.353-5

> ABSTRACT: *We preview our effort, currently underway, to develop theory on the development of community images for new information technologies and on the role these images play in the adoption, diffusion, and implementation of those technologies.*
> REF: 6

(Redwine 1984) Redwine, Samuel T., et al, "DoD Related Software Technology Requirements, Practices, and Prospects for the Future*."* Technical Report IDA Paper P-1788, Institute for Defense Analysis, June, 1984.

> **Category**: technology transfer
> **Key Words**: technology maturation, case study
> **Abstract/Summary:** *This study, funded by the STARS JPO, considers the maturation process for software technologies, including Unix and Smalltalk-80. While the study is not rigorous, it does provide some general*

maturation characteristics and good case studies.
Referenced by (Przbylinski 1988)

(Rice 1982) Rice, Ronald E., Bonnie McD. Johnson, and Everett M. Rogers. Facilitation Adoption of New Office Technology.1982 *Office Automation Digest*: 645-652, April, 1982
    Category: innovation
    Key Words: innovation adoption
    Abstract/Summary: *Building on Rogers' previous work, this paper discusses a five stage model of the innovation process: agenda-setting, matching, redefining, structuring and interconnecting:*
Referenced by (Przbylinski 1988)

(Riddle 1984) Riddle, William E. "The Magic Number Eighteen Plus or Minus Three: A Study of Software Technology Maturation," *ACM SIGSOFT Software Engineering Note* 9 (2):pp. 21-37, April, 1984.
    Category: technology transfer
    Key Words: technology maturation
    Abstract/Summary: *This paper was extracted frorn the Redwine study.*
Referenced by (Przbylinski 1988)

(Roberts 1981) Roberts, Edward B. and Alan R. Fusfeld "Staffing the Innovative Technology-Based Organization.," *Sloan Management Review* :19-34, Spring, 1981.
    Category: innovation
    Key Words: innovation roles, transfer planning
    Abstract/Summary: *In addition to discussing the roles in the innovation process, this paper includes a multi-stage view of a technical innovation project. The authors provide insights into possible implementations of each stage.*
Referenced by (Przbylinski 1988)

(Robertson 1987) Robertson, Thomas S. and Hubert Gatignon, "The Diffusion of High Technology Innovations: A Marketing Perspective," in Johannes M. Pennings and Arend Buiten-dam (editors), *New Technology as Organizational Innovation: The Development and Diffusion of Microelectronics,* chapter 8, pages 179-196. Ballinger Publishing Company, Cambridge, MA, 1987.
    Category: innovation
    Key Words: diffusion research, technology marketing

    Abstract/Summary: *In this article the authors attempt to combine results from diffusion research from the disciplines of marketing and organizational behavior "to derive an enriched model for the study of technology diffusion". They argue that traditional diffusion research ignores supply-side factors, such as the competitive and marketing actions of innovation suppliers. In addition, most existing results do not study contextual variables (e.g., industry competitiveness, return on investment, and industry structure) in great enough depth. The paper goes on to list supply-side and contextual factors affecting diffusion and contains a number of propositions for further study.*
Referenced by (Przbylinski 1988)

(Rogers 1977) Rogers, Everett M., Linda Williams and Rhonda B. West,.*Bibliography of the Diffusion of Innovation,*. Bibliography Council of Planning Librarians Exchange Librarians Number 1420-1422, Institute for Communication Research, Stanford University, December, 1977.
    Category: innovation
    Key Words: diffusion of innovations, Diffusion bibliography
    Abstract/Summary: *This bibliography documents the collection of the Diffusion Documents Center at Stanford University. At the time this was published the center contained approximately 2750 diffusion references. Two types of publications are contained: (1) empirical diffusion studies and (2) non-empirical publications, which include bibliographies, summaries of diffusion findings reported in other publications and theoretical writings.*
Referenced by (Przbylinski 1988)

(Rogers 1981) Rogers, Everett M. and D. Lawrence Kincaid,*Communication Networks: Toward a New Paradigm for Research,* The Free Press, New York, 1981.
    Category: communication
    Key Words: communication network analysis
    Abstract/Summary: *Rogers and Kincaid discuss their paradigm for communication network analysis. Their methods can help transfer organizations track the effects of their dissemination efforts.*
Referenced by (Przbylinski 1988)

(Rogers 1983) Rogers, Everett M., *Diffusion of lnnovation, F*ree Press. New York 1983

Category: innovation
Key Words: innovation diffusion
Abstract/Summary: *Rogers' work is the basis upon which most existing diffusion of innovations work is built. It is a highly readable work that can provide insights into technology transfer planning.*
Referenced by (Przbylinski 1988)

(Roland 1980) Roland, Ronald J., "An Interactive Decision Support System for Technology Transfer Pertaining to Organization and Management," Technical Report AD-A089968, Naval Postgraduate School, July; 1980.
Category: technology transfer
Key Words: decision support systems
Abstract/Summary: *This report more fully describes Roland's DSS for technology transfer of management practices.*
Referenced by (Przbylinski 1988)

(Roland 1982) Roland, Ronald J. "A Decision Support System Model for Technology Transfer.," *Journal of Technology Transfer* 7(1):73-93, 1982.
Category: technology transfer
Key Words: transfer models, transfer aids
Abstract/Summary: *This paper, a short version of Roland's technical report from the Naval Postgraduate School, briefly describes an intelligent system that helps in the design of decision support systems. The prototype was built using the EMYCIN production rule system used at Stanford University. It embodies the linker concepts investigated by Creighton et al.*
Referenced by (Przbylinski 1988)

(Saga 1994) Saga-VL; Zmud-RW, "The nature and determinants of IT acceptance, routinization, and infusion," in Levine, Linda, ed., proceedings of the IFIP TC8 Working Conference on Diffusion, Transfer and Implementation of Information Technology, Software Engineering Institute, Carnegie Mellon Institute, Pittsburgh, PA, North Holland, Amsterdam,, London, New York, Tokyo, 1994.
SOURCE: IFIP-Transactions-A-(Computer-Science-and-Technology). vol.A-45; 1994; p.67 -86

ABSTRACT: *Although it is well recognized that the post-implementation behaviors, e.g., the acceptance, routinization, and infusion of information technology (IT), are critically important to attaining IT implementation success, the dynamics which exist between these behaviors are not, as yet, fully understood. Further, these behaviors have not been deeply grounded within a theoretical foundation, nor have commonly-accepted definitions been developed. This paper through an extensive review of the research literature dealing with post-adoption IT implementation behavior, institutionalization and organizational learning integrates what is currently known about post-adoption behaviors to provide definitions of the constructs, and a set of causal models which theoretically link the constructs to one another as well as to other variables understood to significantly influence IT implementation success.*
REF: 59

(Saxena 1994) Saxena, K.B.C.; Tam, M.M.C.; Chung, W.W.C.; Yung C..L; Ma, L.C.K.; David A.K., "Institutionalization Of Decision Support Technologies In Small Manufacturing Enterprises Of Hong Kong," in Levine, Linda, ed., proceedings of the IFIP TC8 Working Conference on Diffusion, Transfer and Implementation of Information Technology, Software Engineering Institute, Carnegie Mellon Institute, Pittsburgh, PA, North Holland, Amsterdam,, London, New York, Tokyo, 1994.
SOURCE: IFIP-Transactions-A-(Computer-Science-and-Technology). vol.A-45; 1994; p.139-58

ABSTRACT: *Small manufacturing enterprises in Hong Kong are getting increasingly globalized and therefore need to use decision support tools/technologies to remain competitive. As they lack the expertise to deploy these tools/technologies, they either avoid their use or fail in their successful use or institutionalization. A number of approaches have been suggested for institutionalization but many of them provide only the critical success factors and not the dynamics of the institutionalization process. The authors suggest a process-oriented strategic framework for institutionalization of these tools/technologies, which identifies critical factors for successful institutionalization. Finally, they describe a case study where the framework was applied and was successful.*
REF: 50

(Scacchi 1987) Scacchi, Walt and James Babcock, "Understanding Software

Technology Transfer. Non-Proprietary," Technical Report STP-309-87, MCC Report, October, 1987.

Category: technology transfer

Key Words:: transfer bibliography, transfer strategies

Abstract/Summary: *This report synthesizes the current state of the art and practice in software technology transfer, drawing heavily on existing empirical studies. It contains an extensive reference list which was the source for many of the references included in this bibliography.*

Referenced by (Przbylinski 1988)

(Scacchi 1988) Scacchi, Walt, "Understanding Software Technology Transfer: Barriers to Innovation Engineering," TH0218-8/88/0000/0130 *IEEE* pp. 130-135, 1988.

REF: 24

(Schein 1983) Schein, Edgar H., "Corporate Culture: What It Is and How To Change It*," ,*Invited address delivered to 1983 Convocation of the Society of Sloan Fellows, MIT, October 14, 1983 ONR TR 26, Sloan School of Management, November, 1983.

Category: organizational change

Key Words: change agents

Abstract/Summary: *This speech discusses the process of organizational change. It includes a lengthy reference list.*

Referenced by (Przbylinski 1988)

(Schneider 2000) Schneider, Thomas, "Information Theory Primer" www.LECBNCIFCRF.gov~toms/paper/primer,

Category: Information Theory Introduction

Key words: Uncertainty, Shannon, Rate, Bit, Noise

Abstract/Summary: *This primer is written for molecular biologists who are unfamiliar with information theory. Its purpose is to introduce you to these ideas so that you can understand how to apply them to binding sites (1, 2, 3, 4, 5, 6, 7, 8, 9). Most of the material in this primer can also be found in introductory texts on information theory. Although Shannon's original paper on the theory of information (10) is sometimes difficult to read, at other points it is straight forward. Skip the hard parts, and you will find it enjoyable. Pierce later published a popular book (11) which is a great introduction to information theory. Other introductions are*

listed in reference (1). A workbook that you may find useful is reference (12). Shannon's complete collected works have been published (13). Information about ordering this book is given in http://www.lecb.ncifcrf.gov/~toms/bionet.info-theory.faq.html#REFERENCES-Information_Theory.

REF: 15

(Schon 1963) Schon, Donald A., "Champions for Radical New Inventions," *Harvard Business Review,* pp. 77-86, March-April, 1963.

Category: innovation

Key Words: product champion

Abstract/Summary: *This papers summarizes a study conducted by Arthur D. Little Inc., under a contract administered by the National Inventors Council supported by the military services. It provides information on why inventors fail and suggests patterns for success, based on the concept of product champions.*

Referenced by (Przbylinski 1988)

(Schon 1967) Schon, Donald A. *Technology and Change: The New Heraclitus.*, Delacorte Press, New York NY, 1967.

Category: organizational change

Key Words: resistance to change, change agents

Abstract/Summary: *Schon discusses an organizations natural ambivalence to change. Firms must both resist and espouse innovation. The first chapter includes some general definitions of technology and innovation that embrace those of Rogers and others but are much more understandable.*

Referenced by (Przbylinski 1988)

(Staw 1987) Staw, Barry M. and Jerry Ross. "Knowing When To Pull The Plug," *Harvard Business Review* :68 74, March/April, 1987.

Category: technology management

Key Words: resource allocation

Abstract/Summary: *This recent HBR article discusses how to kill development projects, i.e., when rationality should rule over emotional attachment.*

Referenced by (Przbylinski 1988)

(Taylor 1983) Taylor, Bruce. J., "Patterns of Technology Transfer in a Development Group," in *IEEE Computer Society Workshop on Software Engineering Technology Transfer,* Pages 94-98. IEEE Computer Society,

Konover Hotel, Miami Beach FLLL, April 25-27, 1983.

Category: technology transfer

Key Words: tool transfer

Abstract/Summary: *Taylor discusses the use of toolsmiths as a technology transfer mechanism in a Unix environment. While this method can be highly successful, he points out the disadvantages for management .*

Referenced by (Przbylinski 1988)

(Tornatzky 1983) Tornatzky, Louis a. et al. "The Process of Technological Innovation: Reviewing the Literature," Technical Report, National Science Foundation,May, 1983.

Category: innovation

Key Words: innovation diffusion, bibliography

Abstract/Summary: *This extensive NSF study is must reading for those interested in the management of innovation. It summarizes much existing work, while also comparing research across disciplines. I t includes a forty page reference list.*

Referenced by (Przbylinski 1988)

(Tushman 1979) Tushman, Michael L." Managing Communication Network in R&D Laboratories," *Sloan Management Review* 20:37-49, Winter, 1979.

Category: communication

Key Words: communication networks, boundary spanners

Abstract/Summary: *This paper continues the work started by Allen Tushman discusses his contingency model managing communication in R&D , which includes the concept of boundary spanners.*

Referenced by (Przbylinski 1988)

(Tushman 1982) Tushman, Michael L, Columbia University Graduate School of Business, Cambridge, MA. *Readings in the Management of Innovation,*1982.

Category: innovation

Key Words: management of innovation

Abstract/Summary: *This book contains reprints of many pertinent articles, mostly from the Sloan Management Review. A number of references in this bibliography were reprinted here.*

Referenced by (Przbylinski 1988)

(Twiss 1980) Twiss, Brian C. *Managing Technological Innovation.* Longman, New York, 1980

Category: innovation

Key Words: technology management

Abstract/Summary: *This book takes a pragmatic approach to technology management, including chapters on financial evaluation of R&D projects, organization for innovation and technology forecasting.*

Referenced by (Przbylinski 1988)

(Warner 1974) Warner, Kenneth E. "The Need for Some Innovative Concepts of Innovation: An Examination of Research on the Diffusion of Innovations.," *Policy Sciences* 5, pp. 33-451, 1974.

Category: innovation

Key Words: diffusion research

Abstract/Summary: *Warner discusses the definitional problems which are the basis for inconsistencies in diffusion research performed by different disciplines. H e feels that much basic conceptualization and theorizing must be performed before*
research can move
forward.

Referenced by (Przbylinski 1988)

(White 1975) White, Michael J. *Management Science in Federal Agencies. The Adoption and Diffusion of a SocioTechnical Innovation.* Lexington Books, Lexington, MA, 1975.

Category: innovation

Key Words: management of innovation, organizational change

Abstract/Summary: *This book documents White's study of the insertion of management science technology into federal agencies. Management science* is *similar to software technology in that it also has both tool and process components. White includes a detailed model of the organizational responses/changes caused by new technology.*

Referenced by (Przbylinski 1988)

(Wright 1969) Wright, Philip. "Government Efforts to Facilitate Technical Transfer," in William H. Gruber and Donald G. Marquis (editors), *Factors in the Transfer of Technology,* chapter 14, pages 238-251. The M.I.T. Press, Cambridge, MA, 1969.

Category: technology transfer

Key Words: transfer evaluation

Abstract/Summary: *This paper documents the study of NASA's technology transfer efforts. While NASA is often touted as*

an example of effective technology transfer, this study can provide no hard evidence.

Referenced by (Przbylinski 1988)

(Wylie 1982) Wylie, C. Dennis and Robert R. Mackie. "Factors Influencing Organizational Acceptance of Technological Change" in *Training.Final Technical* CRG-TR-82-018, Canyon Research Group, Inc., October, 1982.

Category: organizational change

Key Words: innovation adoption, adoption models

Abstract/Summary: *This report, funded by the Organizational Effectiveness Research Group at the Office of Naval Research, contains a good literature review on the factors in innovation acceptance. It also includes a predictive model of organizational acceptance based mostly on communication and motivational factors .*

Referenced by (Przbylinski 1988)

REFERENCES SECOND ORDER:

(Abernathy 1978) Abernathy, W. J. and Utterbeck, J. M., "Patterns of Industrial Innovation," *Technology Review*, June-July 1978, pp.40-47. Referenced by (Fichman 1993) (Buxton 1991)

(Adler 1990) Adler, P.S. & Shenhard, A. "Adapting Your Technological Base: the Organizational Challenge," Sloan Management Review 32 (1), pp. 25-37, 1990. Referenced by (Fowler 1993)

(Adler 1990) Adler, R S.& Shenbar, A. (1990). "Adapting your technological base: The organizational challenge". *Sloan Management Review* 32(l), 25-37. Referenced by (Fowler 1994)

(Arthur 1987) W.B. Arthur, "Competing Technologies: An Overview," in *Technical Change and Economic Theory*, ed. G. Dos,: Columbia University Press, New York, 1987. Referenced by (Fichman 1993) (Fichman 1994)

(Atre 1990) Atre, S., "The Scoop on OOPS," *Computerworld*, 17 September 1990, pp. 1115-1116. Referenced by (Fichman 1993)

(Ayres 1969) Ayres, Robert U., *Technological Forecasting And Long-Range Forecasting*.McGraw-t.s. Hill Book Company, New York, 1969. Referenced by (Jaakkola 1995)

*(*Babcock 1987*)* Babcock, J., and Scacchi, W., "Understanding Software Technology Transfer," Tech. Report, STP-309-87, Microelectronics and Computer Technology Corp., Austin, Texas, Oct. 1987. Referenced by (Raghavan 1989)

(Bass 1969) Bass, F.M. (1969), "A New Product Growth Model for Consumer Durables.," *Management Science* 15, pp.215-227. Referenced by (Jaakkola 1995)

(Bergner 1983) Bergner, R. M. and Davis, K. E. (1983). Introduction to Volume 3 of *Advances in Descriptive Psychology*, pp. 3-12 JAI Press.,Greenwich, CT: Referenced by (Jeffrey 1988)

(Berliner 1987) Berliner, P. Zave, *An Experiment in Technology Transfer: PAISley Specification of Requirements for an Undersea Lightwave Cable System*, Proceedings of the 9th Conf. On Software Engineering, Monterey, Calif. April 1987. Referenced by (Huseth 1988).

(Berniker 1991) Berniker, E., "Models Of Technology Transfer: A Dialectical Case Study," *Proceedings of the IEEE Conference: The New International Language,* July 1991. Referenced by (Pfleeger 1999)

(Bikson 1987) Bikson, T.K., "Understanding the Implementation of Office Technology," Tech. Note N-2619-NSF, Rand Corp., Santa Monica, June 1987. Referenced by (Raghavan 1989)

(Blackman 1972) Blackman, A.W. (1972), A Mathematical Model for Trend Forecasts. Technological Forecasting and Social Change 3, pp. 441-452. Referenced by (Jaakkola 1995)

(Bobrow 1985) Bobrow, D., et al, Common Loops: Merging Common Lisp and Object-Oriented Programming", Xerox Palo Alto Research Center, August 1985. Referenced by (Huseth 1988).

(Botkin 1992) Botkin, L& Matthews, J. (1992).Winning combination coming wave of entrepreneurial partnerships between large and small companies. New York: John Wiley & Sons, Inc. Referenced by (Fowler 1994) (Fowler 1992)

(Bouldin 1989) Bouldin, B. (1989). *Agents Of Change: Managing The Introduction Of Automated Tools*. Englewood CIM, N.J., Yourdon Press. Referenced by (Fowler 1994) (Fowler 1993)

(Brooks 1987) Brooks, F. P., "No Silver Bullet: Essence and Accidents of Software Engineering," *IEEE Computer* 20, pp. 10-19., 1987 Referenced by (Fichman 1993)

(Bulkeley 1990) Bulkeley, W. M., "Bright Outlook for Artificial Intelligence Yields to Slow Growth and Big Cutbacks," *Wall Street Journal*, 5 July 1990, pp. B1, B3. Referenced by (Fichman 1993)

(Businessweek 1991) "Software Made Simple," *Business Week*, 30 September 1991, pp. 92-100. Referenced by (Fichman 1993)

(Carnevale 1991) Carnevale, M. L., "DSC Says Software Change Led to Phone Outages," W*all Street Journal*, pp. 5, -10, July 1991. Referenced by (Fichman 1993)

(Conner 1982) Conner, D. R.& Patterson, R. W. (1982). "Building Commitment To Organizational Change," *Divining and*

Development Journal, pp. 18-30, April 1982. Referenced by (Fowler 1994)

(Conner 1982) Conner, D.R. & Patterson, R.W., "Building Commitment to Organizational Change," *Training and Development Journal,* pp. 18-30,April 1982. Referenced by (Fowler 1993)

(Davis 1981) Davis, K. E. Introduction to Volume 1 of *Advances in Descriptive Psychology* , pp. 3-11, 1981. Referenced by (Jeffrey 1988)

(Deal 1991) Deal. T. E.,. "Developing A Quality Culture," In R. H. Kilmann (Ed.), *Making Organizations Competitive* , pp. 156175, Jossey-Bass., San Francisco, Cal: 1991. Referenced by (Fowler 1994) (Fowler 1993)

(Dean 1997) Dean, J.J. *Deciding To Innovate. How Firms Justify Advanced Technology.* Cambridge, MA: Ballinger Publishing Company, 1997. Referenced by (Fowler 1994) (Folwer 1993)

(Downs 1976) Downs, G.W., & Mohr, L.B. "Conceptual Issues In The Study Of Innovation," *Administrative Science Quarterly* 21, pp. 700-714, 1976. Referenced by (Fowler 1994) (Fowler 1993)

(Easingwood 1981) Easingwood C., Majahan V., Muller E. *A Non Symmetric Responding Logistic Model for Forecasting Technological Substitution. Technological Forecasting and Social Change* 20, pp. 199-213, 1981. Referenced by (Jaakkola 1995)

(Eutelis 1993) Eutelis Consult, "Scenario Mobile Communications 2010: Study on Forecast Development and Future Trends in Technical Development and Commercial Provision up to the year 2010". *Report to the Commission of the European Communities.* CEC Contract No.48166, 1993. Referenced by (Jaakkola 1995)

(Eveland 1990) Eveland, J.E. and Tornatzky, L.G., "The Deployment of Technology," in *The Processes of Technological Innovation*, eds. L.G. Tornatzky and M. Fleischer (Lexington, Massachusetts: Lexington Books, 1990), pp. 117-148. Referenced by (Fichman 1993)

(Fagan 1976) Fagan, M. "Design And Code Inspections To Reduce Errors In Program Development," *IBM Systems Journal* 15(3), 182-211, 1976. Referenced by (Fowler 1994) (Fowler 1993)

(Farrell 1987) Farrell, J. and Saloner, G., "Competition, Compatibility, and Standards: The Economics of Horses, Penguins, and Lemmings," in *Product Standardization and Competing Strategy*, ed. N.L. Gabel Elsevier Science, Amsterdam: North-Holland, 1987. Referenced by (Fichman 1993)

(Fichman 1992) Fichman, R. and Kemerer, C., "Object-Oriented and Conventional Analysis and Design Methodologies: Comparison and Critique," *IEEE Computer* 25, pp.20-39. 1992. Referenced by (Fichman 1993)

(Fichman 1992) Fichman, R.G. "Information Technology Diffusion: A Review of Empirical Research,". In DeGross, Becker, Elain (Eds.), *Proceedings of the 13th International Conference on Information Systems*, Dallas. pp. 195-206, 1992. Referenced by (Jaakkola 1995)

(Fichman 1993) Fichman, R.G. & Kemerer, C., "Adoption of Software Engineering Innovations: The Case of Object Orientation," *Sloan Management Review*, Winter, pp. 7-22, 1993. Referenced by (Fowler 1993) (Fowler 1994)

(Fichman 1997) Fichman, R.G. and C. F. Kemerer, "The Assimilation Of Software Process Innovations: An Organizational Learning Perspective," *Management Science*, 43 (10), pp. 1345-1363, October 1997. Referenced by (Pfleeger 1999)

(Fisher 1972) Fisher, J.C., Pry, R.H. (1972) *A Simple Substitution Model of Technological Change. Technological Forecasting and Social Change* 3, 7 5-78. Referenced by (Jaakkola 1995)

(Floyd 1968) Floyd A. (1968), "Trend Forecasting: A Methodology for Figures of Merit,'. In Bright J. (cd.), *First Annual Technology and Management Conference.* Prentice Hall Inc . Referenced by (Jaakkola 1995)

(Fowler 1990) Folwer P. & Rifkin, S., "Software Engineering Process Group Guide," Technical Report CMU/SEI-90-TR-24; ESD-90-TR-225, ADA235784), Software Engineering Institute, Pittsburgh, Pa. 1990. Referenced by (Fowler 1993)

(Fowler 1990a) Fowler, P., "Technology Transfer As Collaboration: The Receptor Group." *Proceedings Of The 12th International Conference On Software Engineering* (pp. 332-333). Nice, France, U..S., IEEE Computer Society Press, 1990. Referenced by (Fowler 1993) (Fowler 1994)

(Fowler 1992) Fowler, P. & Levine, L. "Toward A Problem Solving Approach To Software Technology Transition.," In J. Van Lemwen (Ed.), *Proceedings of the IFIP 12th World Computer Congress*, vol. I (pp. 57-64).

Elsevier Science Publishers, Madrid, Spain, The Netherlands: North Holland, 1992. . Referenced by (Fowler 1993) (Fowler 1994)

(Fowler 1992A) Fowler, P. & Maher, J. "Foundations for systematic software technology transition.," *Software Engineering Institute Technical Review '92*, 1-32, 1992. Referenced by (Fowler 1993) (Fowler 1994)

(Freeman 1983) Freeman, P., "Software Engineering: Strategies for Technology Transfer," *Proc. Int'l computing Symp. Applications Systems Development*, pp.333-351, ACM, Berichte, West Germany, 1983,. Referenced by (Raghavan 1989) (Freeman 1988)

(Freeman 1987) Freeman, Peter., *Essential Elements of Software Engineering Education Revisited*, In Software Engineering Education, pp. 61-74, Springer-Verlag, 1987. Referenced by (Freeman 1988)

(Glass 1998) Glass, Robert and Alan Howard, *Software development state-of-the-practice*, Managing System Development, June 1998. Referenced by (Pfleeger 1999)

(Glynn 1990A) Glynn, G. (1990). Semi-formal process model for technology transfer Proceedings of the 12th International Conference on Software Engineering (pp.334-335). Nice, France, U.S.: IEEE Computer Society Press. Referenced by (Fowler 1993) (Fowler 1994)

(Gore 1990) Lavaraj U.A., Gore A.P. (1990) On Interpreting Probability Distributions Fitted to'Firnes of First Adoption. 7L-chnological Forecasting and Social Change 37, 355-370. Referenced by (Jaakkola 1995)

(Grady 1997) Grady, R.B. & Caswell, D.L. *Software Metrics: Establishing A Company Wide Program* Englewood Cliff, NJ: Prentice-Hall, Inc., 1997. Referenced by (Fowler 1993) (Fowler 1994)

(Griss 1997) Griss, Martin and Marty Wasser, Quality Time column, IEEE Software, January 1995. Lai, V. S. and J. L. Guynes, *An assessment of the influence of organizational characteristics on information technology adoption decision: a discriminative approach*, IEEE Transactions on Engineering Management, 44(2), pp. 146-157, May 1997. Referenced by (Pfleeger 1999)

(Hasingwood 1983) Hasingwood C., Majahan V, Muller E. (1983), "A Nonuniform Influence Innovation Diffusion Modelof New Product Acceptance," *Marketing Science* 2, pp. 273-296. Referenced by (Jaakkola 1995)

(Herman 1992) N. D. Herman and T. D. Schneider. "High Information Conservation

Implies That At Least Three Proteins Bind Independently To F Plasmid *Incd* Repeats,' *J. Bacteriol.*, 174: pp. 3558-3560, 1992. Referenced by (Schneider 2000)

(Humphrey 1988) Humphrey, W.S. (1988). "Characterizing The Software Process, " *IEEE Software* 5, pp. 73-79. Referenced by (Fowler 1993) (Fowler 1994)

(Humphreys 1989) Humphreys, W.S.: *Managing The Software Process*, (Addison-Wesley, 1989) Referenced by (Buxton 1991)

(Jaakkola 1991) Jaakkola, Hannu (1991) *An Analysis of the Diffusion of Information Technology in Finnish Industry. Tampa University of Technology,* Publications 70 (In Finnish, English Summary), Tampa. Referenced by (Jaakkola 1995)

(Jaakkola 1991A) Jaakkola, Hannu , "Managing By Modeling –How to Manage the Diffusion of a New technology.". *Proceedings of the Portland International Conference on Management of Engineering and Technology*, IEEE, 1991. Referenced by (Jaakkola 1995)

(Jaakkola 1992) Jaakkola, Harmu "Modifications and Integration of Diffusion Models., " *Proceedings of the Joint EURO XWFIMS X) Qa Conference*, Helsinki., 1992 Referenced by (Jaakkola 1995)

(Jaakkola 1992A) Jaakkola, Hann, "Data models from Theory to Practice - Modeling the Adoption of New Data Models," In Olisup S. (et al.), *Information Modeling and Knowledge,* Bases 111, IOS Press, Amsterdam. pp. 675-690, 1992. Referenced by (Jaakkola 1995)

(Jaakkola 1992B) Jaakkola, Hannu "The Diffusion of Information Technology in Finnish Industry," In Vuon S., Yid-Anttila P. (eds.), *Mastering Technology Diffusion - The Finnish Experience,* ETLA Series B82-1992, Helsinki. pp. 379-432, 1992. Referenced by (Jaakkola 1995)

(Jaakkola 1993) Jaakkola, Hannu, "Diffusion Models for Technology Management,". In Clark J.D., Troxell W.O. (eds.), *Designer Competitiveness. Proceedings of the International Conference on Technology Management* (April 4-7, 1993), Manufacturing. g Excellence Center Press, Denver, Colorado. PP 351-366, 1993. Referenced by (Jaakkola 1995)

(Jaakkola 1993A) Jaakkola, Ilannu, "Toward the Heuristic Model of Technology Diffusion," In the *Proceedings of the IFAC World Congress*, Preprint Vol. 3, Sydney Australia, International Federation for

Automation and Control (IFAC). pp. 441-446, 1993. Referenced by (Jaakkola 1995)

(Jaakkola 1993B) Jaakkola H, Tenhunen H.) "The Impact of Information Technology on Finnish Industry: A Survey of Two Studies," *OECD STI Review* 12, , pp. 53-80, April 1993. Referenced by (Jaakkola 1995)

(Jaakkola 1994) Jaakkola, Hannu "The Heuristic Model of Technology Diffusion.," In Khalil T.M. et al. (eds.) *Management of Technology IV- The Creation of Wealth. Proceedings of the Fourth International Conference on Management of Technology* ,Miami Feb. 27 - March 4, 1994,. Industrial Engineering and Management Press, Institute of Industrial Engineers, Norcrosss, Georgia. 1994. Referenced by (Jaakkola 1995)

(Jaakkola 1994A) Jaakkola, Hannu "The Heuristic Analysis of the Diffusion,". In *Management in Transition: Engineering a Changing World. IEEE Proceedings of the IEEE International Engineering Management Conference 1994*, Oct. 17-19, Dayton, Ohio, 1994. IEEE, New Jersey, s. 123-130, 1994 Referenced by (Jaakkola 1995)

(Jaakkola 1995) Jaakkola, Hannu (1995), "Technology Management with Mathematical Diffusion Models," Working Draft, not published. Referenced by (Jaakkola 1995)

(Jaakkola 1995A) Jaakkola, Hannu (1995a), "Flexibility in Mathematical Diffusion Models," Working Draft, not published. Referenced by (Jaakkola 1995)

(Jeffrey 1983) Jeffrey, H. J., and Putnam, A. O. "The MENTOR Project: Replicating the Functioning of an Organization," In K. E. Davis and R. Bergner (Eds.), *Advances in Descriptive Psychology* (Vol. 3, pp. 243-270) Greenwich, CT: JAI Press, 1983. Referenced by (Jeffrey 1988)

(Kanter 1983) Kanter, R. M. *The Change Masters*: Simon and Shuster. New York, 1983. Referenced by (Fowler 1993) (Fowler 1994)

(Katz 1986) Katz, M.L. and Shapiro, C., "Technology Adoption in the Presence of Network Externalities," *Journal of Political Economy* 94 (1986): 822-841. Referenced by (Fichman 1993)

(Kernighan 1988) Kernighan, B., Mashey, J., "The Unix Programming Environment", *Interactive Programming Environments*, eds. Barstow, Shrobe, Sanewall, McGraw-Hill., Referenced by (Huseth 1988).

(Klein 1993) Klein, M. et al., *A Practitioner's Handbook for Real-Time Analysis: Guide to Rate Monotonic Analysis for Real-Time Systems*, Kluwer Academic Publishers,Boston, Mass, 1993

(Kraemer and Perry 1978) Kraemer, K., and Perry, J., "Innovation Attributes, Policy Intervention and the Diffusing of Computer Applications among Local Governments," *Policy Sciences*, April 1978, pp.179-205 Referenced by (Raghavan 1989)

(Kumar 1992) Kumar, U., Kumar, V. (1992),technological Innovation Diffusion: The Proliferation of Substitution Models and Easing the User's Dilemma. IEFF Transactions on Engineering Management 39,2 (May), 158-168, 1992. Referenced by (Jaakkola 1995)

(Kwon 1987) Kwon, T. H. and Zmud, R.W., "Unifying the Fragmented Models of Information Systems Implementation" in *Critical Issues in Information Systems Research*, eds. J.R. Boland and R. Nirshheim (New York: John Wiley & Sons, 1987). Referenced by (Fichman 1993)

(Leonard-Barton 1988) Leonard-Barton, D, "Implementation as Mutual Adaption of Technology and Organization," *Research Policy* 17, 5 pp. 603-631, 1988. Referenced by (Fichman 1993) (Fowler 1993)

(Leonard-Barton 1988b) Leonard-Barton, D., "Implementation Characteristics of Organizational Innovations," *Communication Research* 15 (1988): 603-631. Referenced by (Fichman 1993)

(Loonard 1988A) Loonard-Barton, D., "Implementation As Mutual Adaptation Of Technology And Organization.," *Research Policy* 17,5 (Oct. 1988), pp.102-110, 1988. Referenced by (Fowler 1994)

(Loonard 1988B) Loonard-Barton, D. "Implementation Characteristics Of Organizational Innovations," *Communication Research* 15. pp. 603-631, 1988.. Referenced by (Fowler 1994)

(Macro 1987) Marco, A., and Buxton, J.N.: *The Craft Of Software Engineering,* Addison-Wesley, 1987 .Referenced by (Buxton 1991)

(Mansfield 1961) Mansfield E.,"Technical Change and the Rate of Imitation," *Econornetrica* 29), pp. 741-765, October , 1961, Referenced by (Jaakkola 1995)

(Martino 1993) Martino J. *Technological Forecasting for Decision Making*, McGraw Hill , Inc. 1993. Referenced by (Jaakkola 1995)

(Martino 1994) Martino J., "An Approach to Balancing the Risks of R&D Performance Goals," in *Management in Transition: Engineering a Changing World.* IEEE, Proceedings of the IEEE International Engineering Management Conference 1994 (Oct. 17-19, 1994) pp.164-167, Dayton, Ohio, IEEE, New Jersey, 1994. Referenced by (Jaakkola 1995)

(Mitchell 1985) Mitchell, T. O. and Davis, K. E. Introduction to Volume 4 of *Advances in Descriptive Psychology* (pp. 3-16). Greenwich, CT: JAI Press, 1985. Referenced by (Jeffrey 1988)

(Moad 1989) Moad, J., "Cultural Barriers Slow Reusability," *Datamation,*, November 1989, pp. 87-92. Referenced by (Fichman 1993)

(Moore 1987) Moore, G. C., "End-User Computing and Office Automation: A Diffusion of Innovations Perspective," *Infor* 25 (1987): 214-235. Referenced by (Fichman 1993)

(Moore 1991) Moore, G. *Crossing the Chasm: Marketing And Selling Technology Products To Mainstream Customers.* Harper-Business. 1991. Referenced by (Fowler 1994) (Fowler 1993)

(Morgan 1986) Morgan, G. (1986). *Images Of Organization.* Beverly Hills, CA.: Sage Publications. Referenced by (Fowler 1994) (Fowler 1993)

(Muller 1990) Majahan, V., Muller, E., Bass, EM. (1990) New Product Diffusion Models in Marketing: A Review and Directions for Research. *Journal of Marketing* 54, 1-26. Majaban V, Muller E., Rajendra K., Srivastva K. (1990a), Determination of Adopter Categories by Using Innovation Diffusion Models. *Journal of Marketing Research* 27, 37-50. Referenced by (Jaakkola 1995)

(Nanotechnology 1994) *Nanotechnology*, 5:1-18, 1994. http://www.lecb.ncifcrf.gov/~toms/paper/nano2/. Referenced by (Schneider 2000)

(Nord 1987) Nord, W.R. & Tucker, S. (1987). *Implementing Routine And Radical Innovations*. Lexington, Mass.: D.C. Heath. Referenced by (Fowler 1994) (Fowler 1993)

(Norton 1987) Norton J., Bass F, "A Diffusion theory Model of Adoption and Substitution of Successive Generations of High Technology Products." *Management Science* 33, pp. 1069-1086, 1987) Referenced by (Jaakkola 1995)

(Orvik 1987) Orvik, J. M., *The Diffusion of Technology Across Cultures*. In K.

E. Davis and A. O. Putnam (Eds.), Advances in Descriptive Psychology (Vol. 6) Greenwich, CT: JAI Press, 1987. Referenced by (Jeffrey 1988)

(Ossorio 1970/1981). Ossorio, P. G. "Outline of Descriptive Psychology for Personality Theory and Clinical Applications." n K. E. Davis (Ed.), *Advances In Descriptive Psychology,* Vol. 1, pp.57-81, JAI Press, Greenwich, CT: 1981. (Originally published in 1970 as *LRI Report* No. 4d. Whittier, CA and Boulder, CO: Linguistic Research Institute.) . Referenced by (Jeffrey 1988)

(Ossorio 1981/1983) Ossorio, P. G. (1981/1983). "A Multicultural Psychology," In K. E. Davis and R. Bergner (Eds.), *Advances in Descriptive Psychology* (Vol. 3, pp. 13-44) Greenwich, CT: JAI Press, 1981. (Originally published in 1981 as *LRI Report* No. 29. Boulder, CO: Linguistic Research Institute.) . Referenced by (Jeffrey 1988)

(Papp 1993) P. P. Papp, D. K. Chattoraj, and T. D. Schneider. "Information Analysis Of Sequences That Bind The Replication Initiator Repa,". *Journal of. Molecular. Biology.*, 233, pp. 219-230, 1993. Referenced by (Schneider 2000)

(Paulk 1991) Paulk, M. et al., "Capability Maturity Model for Software," Technical Report CMU/SEI-91-TR24; ESD-TR-9124: Software Engineering Institute, Pittsburgh, PA. 1991. Referenced by (Fowler 1994) (Fowler 1993)

(Pearl 1925) Pearl, Raymond,*The Biology of Population Growth.*, Knopf, New York, 1925.Referenced by (Jaakkola 1995)

(Pennings 1987) Pennings, J.M., "Technological Innovations in Manufacturing," in N*ew Technology as Organizational Change*, eds. J.M. Pennings and A. Buitendam , pp. 197-216, Ballinger, Cambridge, Massachusetts, 1987. Referenced by (Fichman 1993)

(Perry 1978) Perry, J. and Kraemer, K., "Innovation Attributes, Policy Intervention and the Diffusion of Computer Applications among Local Governments," *Policy Sciences,* pp. 179-205, April 1978. (Raghavan 1989)

(Peterson 1985) Majahan, V., Peterson, R. (1985) Models for Innovation Diffusion. Sage Publications, Beverly Hills. Referenced by (Jaakkola 1995)

(Pfleeger 1997) Pfleeger, Shari Lawrence and Les Hatton*, Investigating The Influence Of Formal Methods*, IEEE Computer, February 1997. 26 Referenced by (Pfleeger 1999)

(Pfleeger 1998) Pfleeger, Shari Lawrence, *Software Engineering: Theory and Practice*, Prentice Hall, Englewood Cliffs, New Jersey, 1998. Referenced by (Pfleeger 1999)

(Pierce 1980) J. R. Pierce. *An Introduction to Information Theory: Symbols, Signals and Noise.* Dover Publications, Inc., New York, second edition, 1980. Referenced by (Schneider 2000)

(Pollak 1993) Pollak, W., "Standards: Leverage for New Technologies,", *Bridge*, Carnegie Mellon University, Software Engineering Institute, Pittsburgh, Pa., 1993. Referenced by (Fowler 1993)

(Premkumar 1995) Premkumar, G. and M. Potter, "Adoption Of Computer-Aided Software Engineering (CASE) Technology: An Innovative Adoption Perspective," *Data Base for Advances in Information Systems* 26 (2-3), pp. 105-124, May-August 1995. Referenced by (Pfleeger 1999)

(Prescott 1995) Prescott, M. B. and S. A. Conger, "Information Technology Innovations: A Classification By IT Locus Of Impact And Research Approach," *Data Base for Advances in Information Systems* 26(2-3), pp. 20-41, May-August 1995. Referenced by (Pfleeger 1999)

(Pressman 1988) Pressman, R, *Making Software Engineering Happen- A Guide for Instituting the Technology*, Prentice Hall, 1988. Referenced by (Hornbach 1988)

(Prieto-Diaz 1987) Prieto-Diaz, Ruben., *A Software Classification Scheme for Reusability*, Software, pp. 6-16, January, 1987. Referenced by (Freeman 1988)

(Przbylinski 1991A) Przybylinski, S., Fowler, P. & Maher, J. (1991, May). "Software Technology Transition.," Tutorial presented at the 13th International Conference on Software Engineering, Austin, TX. Referenced by (Fowler 1994) (Fowler 1993)

(Putnam 1980) Putnam A. O., "Pragmatic Evaluation," *Training and Development Journal*, pp.35-40, May, 1980. Referenced by (Jeffrey 1988)

(Putnam 1981) Putnam A. O. "Communities," In K. E. Davis (Ed.), *Advances in Descriptive Psychology* (Vol. 1 pp. 195-209: JAI Press, Greenwich, CT, 1981. Referenced by (Jeffrey 1988)

(Raghavan 1989) Raghavan, S.A., and Chand, D.R.: 'Diffusing Software Engineering Methods', *IEEE Software*, (4), pp.81-90, 1989. Referenced by (Buxton 1991)

(Raghaven 1986) Raghaven, S, and Chand, D., *Diffusion of Software Engineering Methods*, Wang Institute of Graduate Studies, November 1986. Referenced by (Hornbach 1988)

(Rai 1995) Rai, A., "External Information Source And Channel Effectiveness And The Diffusion Of CASE Innovations: An Empirical Study,: *European Journal Of Information Systems*, 4(2), pp. 93-102, May 1995. Referenced by (Pfleeger 1999)

(Redwine 1984) Redwine, S. T., Becker, L. G., Marmor-Squires, A.B., Martin, R.J., Nash, S.H., & Riddle, WE. DoD "Related Software Technology Requirements, Practices, And Prospects For The Future," (Technical Report IDA Paper P-1788). Alexandria, VA: Institute for Defense Analysis, 1984. Referenced by (Fowler 1993) (Fowler 1994) (Huseth 1988). (Pfleeger 1999) (Raghavan 1989)

(Redwine 1985) Redwine, S.T. and Riddle, W.E., "Software Technology Maturation," *Proc. Eighth Int'l Conf. Software Eng.* CS Press, Los Alamitos, Calif.. pp. 189-200, 1985. (Raghavan 1989)

Roberts, E. (1991). *Entrepreneurs In High Technology. Lessons From Mff And Beyond* New York: Oxford University Press, 1984. Referenced by (Fowler 1994)

(Rifkin 1988) Rifkin, G. and Betts, M., "Strategic Systems Plans Gone Awry," *Computerworld*, 14, pp. 1,104-105., March 1988. Referenced by (Fichman 1993)

(Roberts 1991) *Entrepreneurs in High Technology: Lessons from MIT and Beyond*, Oxvord University Press, New York, 1991. Referenced by (Fowler 1993)

(Rogan 1995) P. K. Rogan and T. D. Schneider. Using information content and base frequencies to distinguish mutations from genetic polymorphisms in splice junction recognition sites. *Human Mutation*, 6:74-76, 1995. http://www.lecb.ncifcrf.gov/~toms/paper/colonsplice/. Referenced by (Schneider 2000)

(Rogers 1983) (Rogers 1995) Rogers, E. M., *Diffusion of Innovation*, Free Press, New York, 1983, 1995. Referenced by (Buxton 1991) (Fichman 1993) (Raghavan 1989) Raghavan 1988) (Jaakkola 1995) (Pfleeger 1999) (Fowler 1993) (Fowler 1994)

(Rosenberg 1982) Rosenberg, N., "On Technological Expectations," in I*nside the Black Box: Technology and Economics,* Cambridge University Press, New York, 1982. Referenced by (Fichman 1993)

(Rothwell 1985) Rothwell, R. and Zegfeld W.: *Reindustrialization and Technology* (Longman, 1985) Referenced by (Buxton 1991)

(Rothwell 1989) Rothwell, R., and Beesley, M.: "The Importance Of Technology Transfer" in Barber, J., Metcalfe, S., and Porteous, M. (Eds.): *Barriers To Growth In Small Firms,* Chap. 5, Routledge, London, 1989. Referenced by (Buxton 1991)

(Roussel 1991) Roussel, R A., Saad. K. N., & Erickson, T J., *Third Generation R&D ,Managing The Link To Corporate Strategy.*: Harvard Business School Press. Boston, 1991. Referenced by (Fowler 1993) (Fowler 1994)

(Sacco 1988) W. Sacco, W. Copes, C. Sloyer, and R. Stark. *Information Theory: Saving Bits*. Janson Publications, Inc., Dedham, MA, 1988. Referenced by (Schneider 2000)

(Scacchi 1987) Scacchi, W. and Babcock, J., *Understanding Software Technology Transfer*, *Microelectronics and Computer Corporations* Technical Report, October , 1987. Referenced by (Hornbach 1988)

(Scheuing 1989) Scheuing, E., *New Product Management*, Merrill Publishing, Columbus, Ohio, 1989. Referenced by (Raghavan 1989)

(Schneider 1984) Schneider, Thomas, "Information Theory Primer" Ch 2. "The Information Content of Binding Sites on Nucleotide Sequences", University of Colorado, 1984. Referenced by (Schneider 2000)

(Schneider 1986) T. D. Schneider, G. D. Stormo, L. Gold, and A. Ehrenfeucht. "Information Content Of Binding Sites On Nucleotide Sequences." *Journal of Molecular Biology.*, 188: pp. 415-431, 1986. Referenced by (Schneider 2000)

(Schneider 1988) T. D. Schneider. "Information And Entropy Of Patterns In Genetic Switches," In G. J. Erickson and C. R. Smith, editors, *Maximum-Entropy and Bayesian Methods in Science and Engineering*, volume 2, pages 147-154, Kluwer Academic Publishers, Dordrecht, The Netherlands, 1988. Referenced by (Schneider 2000)

(Schneider 1989) T. D. Schneider and G. D. Stormo. "Excess Information At Bacteriophage T7 Genomic Promoters Detected By A Random Cloning Technique," *Nucleic Acids Res.*, 17: pp. 659-674, 1989. Referenced by (Schneider 2000)

(Schneider 1989) T. D. Schneider. *Sequence Logos, Machine/Channel Capacity, Maxwell's Demon, And Molecular Computers: A Review Of The Theory Of Molecular Machines.* Referenced by (Schneider 2000)

(Schneider 1990) T. D. Schneider and R. M. Stephens. "Sequence Logos: A New Way To Display Consensus Sequences," *Nucleic Acids Res.*, 18: pp. 6097-6100, 1990 .http://www.lecb.ncifcrf.gov/~toms/paper/logopaper/. Referenced by (Schneider 2000)

(Schneider 1992) T. D. Schneider. "Sequence Logos, Machine/Channel Capacity, Maxwell's Demon, And Molecular Computers: A Review Of The Theory Of Molecular Machines," *Journal of. Molecular. Biol.*ogy, 228, pp. 1124-1136, 1992. http://www.lecb.ncifcrf.gov/~toms/paper/splice/. Referenced by (Schneider 1984)

(Schoeman 1977) Majahan, V. Schoeman. M.E.F "Gcneralized Model for the Time Pattern of Diffusion Process" IEEE Transactions on Engineering Management EM-24. I (Feb.), 12-18, 1977. Referenced by (Jaakkola 1995)

(Schum 1994) Schum, David A., *Evidential Foundations of Probabilistic Reasoning*, Wiley Series in Systems Engineering, John Wiley, New York, 1994. Referenced by (Pfleeger 1999)

(Shannon 1948) Shannon, C.E., "A Mathematical Theory Of Communication," *Bell System Tech. J.*, 27:379-423, 623-656, 1948.http://cm.bell-labs.com/cm/ms/what/shannonday/paper.html. Referenced by (Schneider 2000)

(Sharif 1976) Sharif M.N., Kabir C., *Systems Dynamics Modeling for Forecasting Multilevel Technological Change. Technological Forecasting and Social Change* 8, pp. 353-364, 1976. Referenced by (Jaakkola 1995)

(Sharif 1981) Sharif, M.N., Ramanathan, K. *Binomial Innovation Diffusion Models with Dynamic Potential Adopter Population. Technological Forecasting and Social Change* 20, pp. 63-9, .1981. Referenced by (Jaakkola 1995)

(Skiadas 1985) Skiadas C. Two Generalized Rational Models for Forecasting Innovation Diffusion.Technological Forecasting and Social Change 27, 39-61, 1985. Referenced by (Jaakkola 1995)

(Sloane 1993) N. J. A. Sloane and A. D. Wyner. *Claude Elwood Shannon: Collected Papers*. IEEE Press, Piscataway, NJ, 1993. Referenced by (Schneider 2000)

(Souder 1987) Souder, W.E., *Managing New Product Innovations.* Lexington Books,

New York, 1987. Referenced by (Fowler 1993) (Fowler 1994)

(Souder 1990) Souder W.E.., Nashar, A.S., & Padmanabban, V. (1M). "A Guide To The Best Technology Transfer Practices," *Journal of Technology Transfer* 15, pp. 5-16, 1990. Referenced by (Fowler 1994)

(Stallman 1986) Stallman, R., *GNU Emacs Manual*, February 1986. Referenced by (Huseth 1988).

(Stephens 1992) R. M. Stephens and T. D. Schneider. Features of spliceosome evolution and function inferred from an analysis of the information at human splice sites. Referenced by (Schneider 2000)

(Stewart 1991) Stewart, M., "Object Projects: What Can Go Wrong," *Hotline on Object-Oriented Technology* 2, pp. 15-17, 1991. Referenced by (Fichman 1993)

(Strauss 1993) Strauss, S., and Ebenau, R.G., *Software Inspection Process*, McGraw Hill, New York, 1993. Referenced by (Fowler 1993)

(Thurow 1992) Thurow, L., "Who Owns the Twenty-First Century?" *Sloan Management Review*, , pp 5-17. Spring 1992, Referenced by (Fichman 1993)

(Tibus 1961) M. Tribus. *Thermostatics and Thermodynamics*.D. van Nostrand Company, Inc., Princeton, N. J., 1961. Referenced by (Schneider 2000)

(Tornatzky 1982) Tonatzky, L.G. and Klein, K., "Innovation Characteristics and Innovation-Adoption Implementation: A Metaanalysis of Findings," IEEE Trans. *Eng. Management*, Jan. 1982, pp.28-45. Referenced by (Raghavan 1989)

(Tornatzky 1983) Tornatzky et al, L.G., "The Process of Technological Innovation: Reviewing the Literature," Tech. Report, Nat'l Science Foundation, Washington, D.C., 1983. Referenced by (Raghavan 1989)

(Tornatzky 1990) Tornatzky, L. G. & Fleischer, M. *The Processes Of Technological Innovation,* Lexington Books, Leximgton, Mass, 1990. Referenced by (Fowler 1993) (Fowler 1994)

(Van De Ven 1991) Van de Ven, A.H., "Managing the Process of Organizational Innovation" in *Changing and Redesigning Organizations*, ed. G.P. Huber Oxford University Press, New York, 1991. Referenced by (Fichman 1993)

(Von Hippel 1988) Von Hippel, E.: *Sources of Innovation,* Oxford University Press, 1988. Referenced by (Buxton 1991)

(Walton 1989) Walton, R.E., *Up and Running*, Harvard Business School Press, Boston, 1989. Referenced by (Fowler 1993)

(Waltozi 1989) Waltozi, R.E. (1989). *Up and Running.* Harvard Business School Press, Boston, 1989. Referenced by (Fowler 1994)

(Willams 1990) Williams, F. & Gibson, DX aids.). *Technology Transfer A Communication Perspective.* Sap Publications, Newbury Park, CA., 1990. Referenced by (Fowler 1993) (Fowler 1994)

(Willis 1983) Willis, R. (1983). "Technology Transfer Takes 6 +1- 2 Years," In R. Morton (FA), *Proceedings of the IEEE* CanWuter Society Workshop on Software Engineering Technology Transfer (pp.108-117). Miami Beach. Fla.: 1EEE Computer Society Press. Referenced by (Fowler 1993) (Fowler 1994) (Raghavan 1989)

(Yourdon 1998) Yourdon, Edward, *Application Development Strategies Newsletter*, February 1998. Referenced by (Pfleeger 1999)

(Zelkowitz 1995) Zelkowitz, Marvin V., *Assessing Software Engineering Technology Transfer Within NASA*, NASA Technical Report NASA-RPT-003095, National Aeronautics and Space Administration, Washington, DC, January , 1995. Referenced by (Pfleeger 1999)

(Zelkowitz 1998) Zelkowitz, Marvin V., Dolores R. Wallace and David Binkley, *Understanding The Culture Clash In Software Engineering Technology Transfer*, University of Maryland Technical Report, 2 June 1998. Referenced by (Pfleeger 1999)

(Zmud 1992) Zmud, R.W. & Apple, L.E., "Measuring Technology Incorporation, Liaison," *Journal of Product Innovation Management* 9, pp.149-155, 1992 Referenced by (Fowler 1993) (Fowler 1994)

THIS PAGE
INTENTIONALLY LEFT BLANK

ANNOTATED BIBLIOGRAPHY CITED REFERENCE DATASET

This bibliography also contains distribution data of all of the research "messages" used in this dissertation and cited in the annotated bibliography. We collected all of the references and determined the references each article cited. We then determined the age of the references relative to the date on the citing publication. Since we think we now how this dissertation will be indexed, we can do a search and determine the input entropy, and a first approximation of β, the amount of feedback or input from the environment. Future research can use this data to calibrate other models and efforts.

This is found on the CD in directory

Entropy data analysis\ Refs_TechTx_Study_Ageing 10 20 ref.xls

THIS PAGE INTENTIONALY BLANK

INDEX

THIS PAGE INTENTIONALLY LEFT BLANK

INITIAL DISTRIBUTION LIST

Dudley Knox Library 2
Naval Postgraduate School
411 Dyer Rd.
Monterey, CA 93943-5000

Defense Technical Information Center 2
8725 John J. Kingman Rd.
Suite 0944
Ft. Belvoir, VA 22060-6218

National Technical Information Service 2
Port Royal Rd
Springfield VA 22161

Dr. Luqi 2
Naval Postgraduate School
Code CS/Lq
411 Dyer Rd.
Monterey, CA 93943-5000

Dr. Valdis Berzins 1
Naval Postgraduate School
Code CS/Bz
411 Dyer Rd.
Monterey, CA 93943-5000

Dr. Dan Boger 1
Naval Postgraduate School
Code CS/Bo
411 Dyer Rd.
Monterey, CA 93943-5000

Dr. John Osumundson 1
Naval Postgraduate School
Code CC
411 Dyer Rd.
Monterey, CA 93943-5000

Dr. Vladin Jovonavic 1
Math and Computer Science Department
Georgia Southern University
Statesboro, GA 30460-8093

Dr. Barry Boehm   1
Computer Science Department
University of Southern California
941 W. 37th Place
Los Angeles, CA 90089-0781

Army Materiel Command   1
Commanding General
ATTN: AMC-CG
General Kern
5001 Eisenhower Ave
Alexandria, VA 22333-5000

Army Materiel Command   1
Deputy Commanding General
ATTN: AMC-DCG
LTG Roy Beauchamp
5001 Eisenhower Ave
Alexandria, VA 22333-5000

Army Materiel Command   1
Deputy Chief of Staff of Research and Development Acquisition
ATTN: AMC-RDA
5001 Eisenhower Ave
Alexandria, VA 22333-5000

U.S. Army Tank-automotive & Armaments Command   1
Commanding General
ATTN: MG N. Ross Thompson, III
Warren, MI 48397-5000

U.S. Army Tank-automotive & Armaments Command   1
Deputy Commander, TACOM
ATTN: Mr. Jerry Chapin
Warren, MI 48397-5000

U.S. Army Tank-automotive & Armaments Command   1
Tank Automotive Research, Development and Engineering Center
Director
ATTN: Dr. Richard McClelland
Warren, MI 48397-5000

U.S. Army Tank-automotive & Armaments Command,   5
Tank Automotive Research, Development and Engineering Center
Associate Director
Next Generation Software Engineering
ATTN: AMSTA-TR-R/265
Warren MI 58397-5000

U.S. Army Tank-automotive & Armaments Command Tank Automotive   2
 Research, Development and Engineering Center
AMSTA-TR-R/265-TIC
ATTN: Technical Information Center
Warren, MI 48397-5000

U.S. Army Tank-automotive & Armaments Command   1
Life Cycle Software Engineering Center
US. Army TACOM-ARDEC
Mr. Dan Nathan
ATTN: AMSTA-AR-FSF-S, B352
Picatinny, NJ 07806-5000

U.S. Army Tank-automotive & Armaments Command   1
US. Army TACOM- ARDEC
ATTN: Library
Picatinny, NJ 07806-5000

Office of the Chief Information Officer/G-6   1
Director of Information Systems DISC4
SAIS-ZA
The Pentagon
Washington DC 20301-1400

US Army Acquisition Executive   1
The Pentagon, Washington DC 20301-1400

Military Deputy Assistant Secretary of Army   1
Acquisition, Logistics, and Technology
LTG John S. Caldwell
The Pentagon, Washington DC 20301-1400

Dr. Anthony J. Tether   1
Director Defense Advanced Research Projects Agency (DARPA)
ATTN: Dr. Anthony J. Tether
3701 North Fairfax Drive
Arlington, VA 22203-1714

Dr. Mikhail Auguston 1
Computer Science Department
New Mexico State University
Las Cruces, NM 88003

Pam Binns 1
Honeywell Technology Center
Honeywell Laboratories
Minneapolis, MN

John Bohn 1
U.S. Army Next Generation Software Engineering (AMSTA-TR-R/265)
U.S. Tank-Automotive & Ammunition Command
Warren, MI 48397-5000

Nimitz Library 1
U.S. Naval Academy
589 McNair Rd.
Annapolis, MD 21402-5029

Laura Malone 1
Software Engineering Institute
Software Engineering Information Repository
Carnegie Mellon University
Pittsburgh, PA 15213-3890

USMA Library 1
United States Military Academy, West Point
Bldg 757
West Point, NY 10996-1799

DARPA 1
DARPA/ITO
3701 Fairfax Drive
Arlington, VA 22203-1714

Dr. William Sander 1
U.S. Army Research Office - AMSRL-RO-IS
P.O. Box 12211
Research Triangle Park, NC 27709-2211

Frank D. Anger 1
The National Science Foundation
4201 Wilson Boulevard
Arlington, VA 22230

Barker Engineering Library   1
Massachusetts Institute of Technology
77 Massachusetts Avenue, 10-500
Cambridge, MA 02139

Kresge Engineering Library   1
University of California, Berkeley
110 Bechtel Engineering Center
Berkeley, CA 94720-6000

The Goddard Space Flight Center Library   1
Code 292, Building 21
Greenbelt, MD 20771

The Sheridan Libraries   1
Johns Hopkins University
3400 N. Charles St.
Baltimore, MD 21218

Academic Library (AFIT/LD)   1
Air Force Institute of Technology
2950 P Street
Building 642 Area B
Wright-Patterson AFB, OH 45433-7765

Cordell Green   1
Kestrel Institute
3260 Hillview Avenue
Palo Alto, California 94304

Dr. Lori Clark   1
Department of Computer Science
University of Massachusetts at Amherst
Amherst, MA 01003

Dr. Leon Osterwell   1
Department of Computer Science
University of Massachusetts at Amherst
Amherst, MA 01003

Dr. David Dampier   1
Department of Computer Science and Eng.
Southern Methodist University
Dallas, Texas 75275-0122

Dr. Michael S. Saboe   10
U.S. Army Next Generation Software Engineering - (AMSTA-TR-R)
U.S. Army Tank-Automotive & Ammunition Command
Warren, MI 48397-5000

Library of Congress   2
Copyright Office
101 Independence Avenue, S.E.
Washington, D.C. 20540